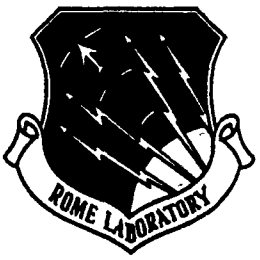
RL-TR-91-106
Final Technical Report
June 1991






# LINEAR MICROCIRCUIT FAULT MODELING AND SIMULATION

David Sarnoff Research Center

Benjamin R. Epstein, Martin H. Czigler, Steven R. Miller



*APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.*



Rome Laboratory
Air Force Systems Command
Griffiss Air Force Base, NY 13441-5700

This report has been reviewed by the Rome Laboratory Public Affairs Office (PA) and is releasable to the National Technical Information Service (NTIS). At NTIS it will be releasable to the general public, including foreign nations.

RL-TR-91-106 has been reviewed and is approved for publication.

APPROVED: E. Paul Ratazzi

E. PAUL RATAZZI
Project Engineer

APPROVED: John J. Bart

JOHN J. BART
Technical Director
Directorate of Reliability & Compatibility

FOR THE COMMANDER: Ronald Raposo

RONALD RAPOSO
Directorate of Plans & Programs

If your address has changed or if you wish to be removed from the Rome Laboratory mailing list, or if the addressee is no longer employed by your organization, please notify RL(RBRA ) Griffiss AFB NY 13441-5700. This will assist us in maintaining a current mailing list.

Do not return copies of this report unless contractual obligations or notices on a specific document require that it be returned.

# REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188



| 1. AGENCY USE ONLY (Leave Blank) | 2. REPORT DATE | 3. REPORT TYPE AND DATES COVERED |
|---|---|---|
| | June 1991 | Final May 89 - Feb 91 |

**4. TITLE AND SUBTITLE**
LINEAR MICROCIRCUIT FAULT MODELING AND SIMULATION

**5. FUNDING NUMBERS**
C - F30602-89-C-0076
PE - 62702F
PR - 2338
TA - 01
WU - 6U

**6. AUTHOR(S)**
Benjamin R. Epstein, Martin H. Czigler, Steven R. Miller

**7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)**
David Sarnoff Research Center
CN5300
Princeton NJ 08543-5300

**8. PERFORMING ORGANIZATION REPORT NUMBER**

**9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)**
Rome Laboratory (RBRA)
Griffiss AFB NY 13441-5700

**10. SPONSORING/MONITORING AGENCY REPORT NUMBER**
RL-TR-91-106

**11. SUPPLEMENTARY NOTES**
Rome Laboratory Project Engineer: E. Paul Ratazzi/RBRA/(315) 330-2946, DSN 587-2946
NOTE: Rome Laboratory (formerly Rome Air Development Center/RADC)

**12a. DISTRIBUTION/AVAILABILITY STATEMENT**
Approved for public release; distribution unlimited.

**12b. DISTRIBUTION CODE**

**13. ABSTRACT** (Maximum 200 words)

Fault analysis in analog microcircuits is useful for tracing the effects of process variations during IC production that lead to circuit failure. Likewise, fault analysis can target specific failure mechanisms that occur during fabrication and field use of ICs. Furthermore, fault analysis is a necessary step in grading the testability of a microcircuit, as well as determining the fault coverage of a specific test suite. This report describes an approach which has aimed to develop systematic methods that can detect faults in analog and mixed-mode ICs by analyzing the response signatures of "good" and "faulted" ICs. In this study, these signatures were obtained from circuit simulations, but empirical data may also be used. Classical multivariate discrimination techniques were used to classify a tested circuit as "good" or "bad", with identification of the most likely fault occurring in the circuit. Calibration of the method relied on extensive circuit simulation under nominal and faulted circuit operation. Nominal variations in the IC components and component/model parameters were accounted for during calibration and fault classification/detection by use of Monte Carlo methods

| 14. SUBJECT TERMS | 15. NUMBER OF PAGES |
|---|---|
| Analog, Linear, Fault, Microcircuit, Failure, Modeling, Simulation, Discrimination, Detection, Classification, Fault Analysis, Integrated Circuit, Monte Carlo, Multivariate | 130 |
| | 16. PRICE CODE |

| 17. SECURITY CLASSIFICATION OF REPORT | 18. SECURITY CLASSIFICATION OF THIS PAGE | 19. SECURITY CLASSIFICATION OF ABSTRACT | 20. LIMITATION OF ABSTRACT |
|---|---|---|---|
| UNCLASSIFIED | UNCLASSIFIED | UNCLASSIFIED | U/L |

# EVALUATION

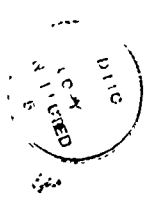
A practical approach to fault modeling and simulation in analog and mixed-mode (analog/digital) circuits has been an elusive goal for many years. Our research (see the Interim Report for this program, RADC-TR-90-30) shows that the Air Force has been looking at this problem intermittently since the 1960s as it applies to electronic board and assembly diagnostics. Other researchers in government, industry, and academia have also been addressing this problem for an equally lengthy period of time. Varied levels of success and practicality have been achieved.

While it is no panacea, and by no means a finished product, this program provides Rome Laboratory with an important step in developing the field of analog fault analysis to a level comparable to that of digital devices. With this capability, there will be a better understanding of the impact analog design techniques have on device and system testability and reliability, as well as added diagnostics capability.

Based on the results of this program, Rome Laboratory is continuing this research with both in-house efforts and contractual programs.

Paul Ratazzi

E. PAUL RATAZZI, RL Project Engineer
Reliability Assurance Branch
Microelectronics Reliability Division



| Accession For | |
|---|---|
| NTIS GRA&I | ☑ |
| DTIC TAB | ☐ |
| Unannounced | ☐ |
| Justification | |
| By | |
| Distribution/ | |
| Availability Codes | |
| Dist | Avail and/or Special |
| A-1 | |

# Table of Contents

# List of Figures

# List of Tables

# Part 1. Methodology

## 1.1. Introduction

The development of methods to analyze faults in analog circuits originally evolved to provide a means of rapidly identifying component failures and assembly error in circuit boards. Unlike fault analysis for digital circuits, analog fault analysis has been comparatively slow to evolve, with only a few practical implementations. Several factors have contributed to making analog circuit fault analysis more of a technical challenge than the digital counterpart:

- Analog circuits have a continuum of possible failure conditions. In contrast, digital circuit failure is dominated by "stuck at" and other "hard" fault conditions that can be analyzed using well-developed systematic procedures;
- A "good" component may be within a prescribed manufacturing tolerance, but not at the exact nominal value of the circuit design;
- Complex feedback and nonlinear circuit operation, which are influenced by faults in a non-linear manner;
- Circuit simulation is slow and subject to numerical difficulties (e.g., convergence, modeling error)
- The exact nature of the fault may not be known; hence, difficult to model;
- A fault in one component can cause faults in other components.

Analog fault analysis covers the procedures of circuit measurement, fault detection, and fault classification. The disciplines of test generation and circuit simulation are also tightly coupled to fault analysis problems. Furthermore, and especially for integrated circuits (ICs), issues on yield prediction and device modeling influence the performance of any fault analysis scheme.

During the 1960s, analog fault diagnosis methods evolved as the result of U.S. Dept. of Defense interest in providing rapid field servicing of circuit boards in weapons, navigation, and communication systems. A number of studies followed, culminating in a large number of theoretical treatments on analog circuit fault analysis during the late 1970s and early 1980s. A comprehensive review on the topic has been given in the Interim Report [19], as well as in other references [5,11,15].

Analog fault detection and classification generally can be divided into two major categories of methodology: *simulation-before-test* and *simulation-after-test.* Broadly speaking, the former covers "fault dictionary" and other pattern matching approaches, while the latter covers the "parameter estimation" and related methods. Simulation-after-test effectively "reverse engineers" a circuit to determine the values of the circuit component parameters. This is performed by taking a series of voltage and/or current measurements of the circuit, then applying numerical analyses to arrive at the component values. The method assumes knowledge of the circuit's topology, and that all component models are sufficiently accurate. Details of simulation-after-test methods can be found elsewhere [5,11,15,19]. Unfortunately, simulation-after-test has severe drawbacks that make such methods impractical for fault analysis in ICs. These drawbacks include:

1) Catastrophic faults cannot be readily simulated without substantially increasing the complexity of the numerical analyses;
2) The component models might not readily account for the true fault mechanisms;
3) The techniques often require large amounts of computation and complicated algebraic relations describing faults in even simple circuits;
4) Many of the techniques fail when nominal component variation is accounted for.
5) Some methods require measurements that may not be practical with ICs (e.g., internal circuit probing and short-circuiting circuit ports to determine currents).

In view of the potential complications encountered with simulation-after-test methods, our approach has been oriented toward the simulation-before-test approach. This approach first determines the circuit responses that are likely to occur, given *a priori* the anticipated types of IC faults to be encountered during IC production or when the ICs are in field use. These responses, usually nodal voltages, can be simulated or measured. The response data are stored in computer files and make up what is commonly known as a *fault dictionary* or *training set.* Next, measurements of ICs under test are made. The measurement data are then compared against the collection of circuit responses from the fault dictionary. A set of responses that best match a given set of measurement data determines the type of fault detected, if present, in the IC. For the purposes of our study, all "measurement" data have been obtained by simulations.

Our study has investigated two approaches to simulation-before-test fault analysis: 1) *fault detection and classification*, and 2) *Go/No-Go testing*. Both approaches are based on similar assumptions and computations. For simplicity, only catastrophic faults are considered in the bulk of this study.

## 1.2. Fault Detection and Classification

This study's approach to fault detection and classification relies on *multivariate discrimination analysis*. We will now review discrimination analysis in the context of how it was used to detect and classify simulated faults in analog and mixed-mode ICs. Simple hypothetical circuits will be used in this discussion. Simulations of real circuits will be discussed in detail in Part 2 of this report.

### 1.2A. Univariate Case

***Univariate voltage distribution***

An understanding of the multivariate statistics used in this study is best grasped by first understanding the univariate case. The word univariate implies that only one measurement variable is considered. Figure 1-1 shows a hypothetical circuit in which $v_1$ (the voltage drop across resistor $R_B$) is the variable measured. We now assume that resistors $R_A$ and $R_B$ are not fixed, but random. That is, the resistor parameters take on statistical distributions to account for variations in manufacturing processes and changes in environmental conditions that affect the resistance values. For the purposes of this study, we will assume that the resistors have normal distributions with prescribed means and standard deviations. (The assumption of normality can be relaxed when nonparametric methods are used, to be discussed later in this report).

Given the inherent variability of the resistors, we must perform a Monte Carlo simulation of the circuit to observe the variability in the measured variable $v_1$. The Monte Carlo simulations carry out a series of N simulations on the circuit, where each simulation randomly selects values of $R_A$ and $R_B$ from the resistance distributions. $v_1$ then takes on a normal distribution with a mean voltage $v_{m1}$ and standard deviation $\sigma_1$. The distribution is described by the relation:

$$f(v_1) = \frac{1}{\sqrt{2\pi\sigma_1^2}} \cdot e^{-\frac{\left[\frac{(v_1 - v_{m1})}{\sigma_1}\right]^2}{2}} \tag{1-1}$$

where the standard deviation $\sigma$ is estimated as:

$$Variance = \sigma_1^2 = \frac{\Sigma (v_1 - v_{m1})^2}{N} \qquad (1\text{-}2)$$

$v_1$ is shown in Figure 1-2. Note that if the circuit were nonlinear (e.g., the resistances were voltage-dependent), the distribution for $v_1$ would not necessarily be of a normal form. However, as we will see in the examples of Part 2, the assumption of normality remains a reasonable approximation even with highly nonlinear circuits. Normality is assumed in such cases to simplify the discrimination mathematics.

***Presence of faults***

Now consider the effect of introducing a fault to the circuit of Figure 1-1. Catastrophic faults in the resistors can be represented by drastic changes in the resistance values where a very low resistance value (relative to nominal) can represent a short circuit, and a very high resistance value can represent an open circuit. Effects on the distribution of $v_1$ are shown in Figure 1-3 as the circuit undergoes catastrophic faulting. Certain fault types cannot be distinguished given only the $v_1$ data. These fault types constitute ambiguity sets. Ambiguity sets can be avoided by a sufficient number of different types of measurements. In the present example, measurement of current through the resistors would help to distinguish the overlapping faults.

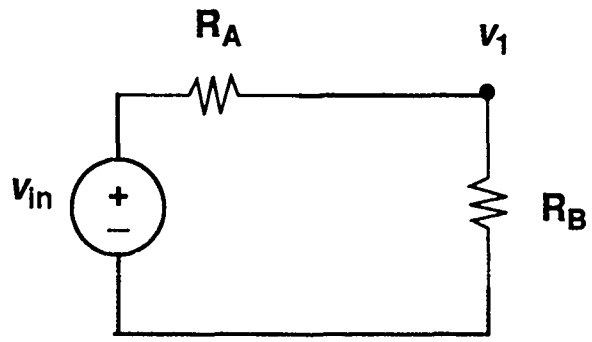



***Figure 1-1.*** Simple circuit illustrating a univariate circuit response. Resistors $R_A$ and $R_B$, which can be nonlinear, are nondeterministic with normal distributions.

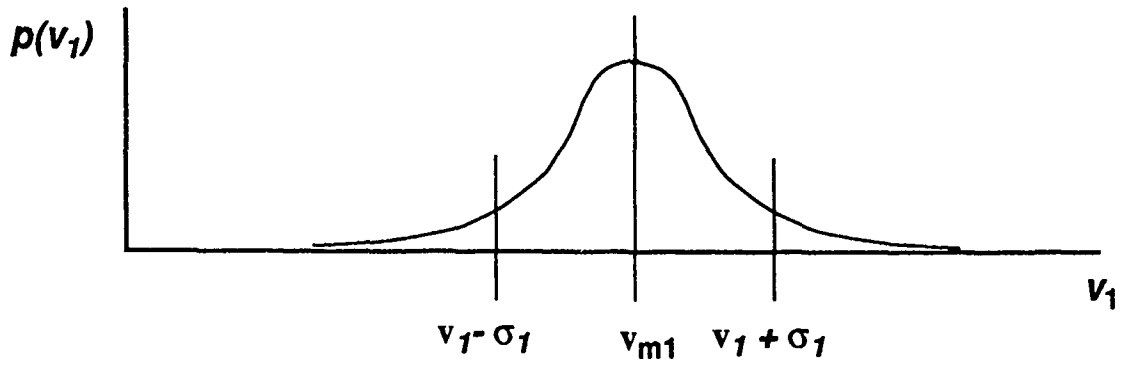



*Figure 1-2.* Normal distribution of circuit voltages from the hypothetical circuit in Figure 1-1. $p(v_1)$ is the probability density function.

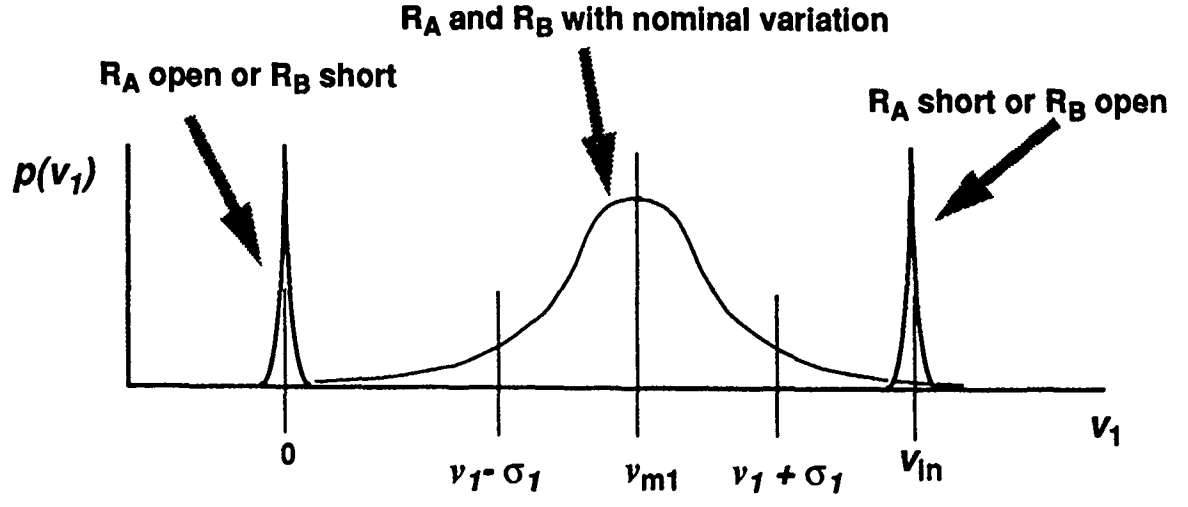



*Figure 1-3.* Distributions of circuit performance arising from faulting resistors in Figure 1-1.

### *Fault classification*

To make the sets of distributions better resemble the types of distributions encountered in more complicated circuits, let us now assume that when a univariate circuit of many possible fault types is faulted, a scalar quantity such as $v_1$ continues to take on normal distributions with shifted means and standard deviations (Figure 1-4). To determine the type of fault present in a given sample circuit, we measure its $v_1$. We then compute the normalized distance of the measured $v_1$ to each of the distributions according to the relation:

$$\left(\frac{v_1 - v_{m1i}}{\sigma_{1i}}\right)^2 = (v_1 - v_{m1i})\,(\sigma_{1i}^2)^{-1}\,(v_1 - v_{m1i}) \qquad (1\text{-}3)$$

Here the subscript $i$ refers to the fault type that the measured $v_1$ is compared to. The fault type corresponding to the smallest normalized distance value is the most likely fault present in the circuit. Note that if the circuit has no faults (i.e., it is nominal), the distribution of $v_1$ corresponding to the nominal circuit would have the closest distance measure to the $v_1$ from the sample circuit. In other words, "good" circuits are classified in a manner identical to the classification of faulted circuits.

An important problem arises when a fault that is introduced to a sample circuit is not accounted for in the training set. In this case, the closest fault type would be selected as the fault present, which would likely be incorrect. One way in which unaccounted faults can be identified as such is simply to assign a threshold to the distance measure of eq. 1-3. Any distance measures exceeding the threshold would then correspond to faults that cannot be identified; meanwhile, the sample circuit would at least be flagged as being faulted.

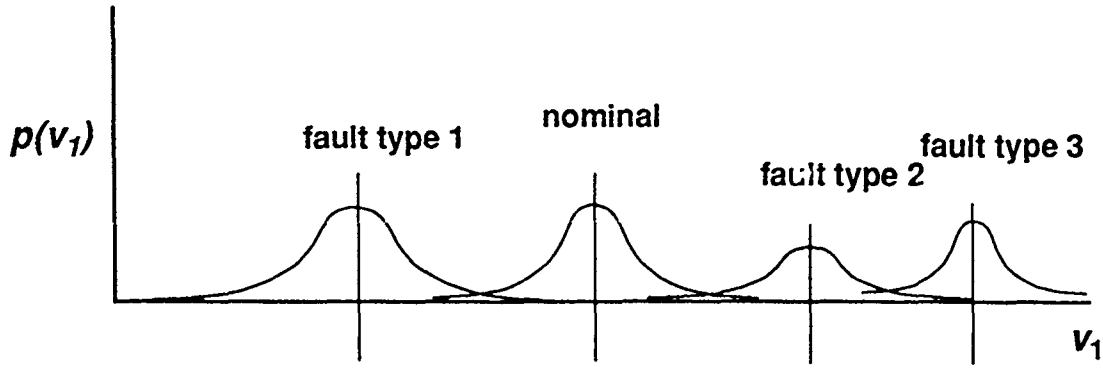



***Figure 1-4.*** Univariate distributions for several fault types affecting a single voltage point v1.

### 1.2B. Multivariate Case

Multivariate statistics are applied when more than one voltage is measured. Therefore, we have to account for the distributions of more than one measurement variable. Figure 1-5 depicts a circuit suitable for a fault analysis of its resistances using multivariate discrimination analysis. In this case, voltages $v_1$, $v_2$, and $v_3$ are measured. Faults can occur by shifting the resistance values, which would result in various combinations of shifts in the distributions (means and standard deviations) of the measured voltages. The distributions of each fault type is given as follows (where the subscript $i$ represents the $i^{th}$ fault type):

$$f_i(V) = \frac{1}{|\Sigma_i|^{\frac{1}{2}} (2\pi)^{\frac{p}{2}}} \cdot e^{-\frac{1}{2}((V-\mu_i)^T \Sigma^{-1} (V-\mu_i))} \tag{1-4}$$

Here $V$ represents a vector of measured voltages. In the present example, vector $V$ is made up of the voltage components $v_1$, $v_2$, and $v_3$. $\mu_i$ is a vector containing the average voltage components over $N$ Monte Carlo runs; $p$ is the size of the vector $V$ (in this case 3). The symbol $\Sigma_i$ is the *covariance matrix* corresponding to the $i^{th}$ fault type. $\Sigma_i$ is estimated as:

$$\begin{bmatrix} E(v_1 - v_{m1})^2 & E(v_1 - v_{m1})(v_2 - v_{m2}) & E(v_1 - v_{m1})(v_3 - v_{m3}) \\ E(v_2 - v_{m2})(v_1 - v_{m1}) & E(v_2 - v_{m2})^2 & E(v_2 - v_{m2})(v_3 - v_{m3}) \\ E(v_3 - v_{m3})(v_1 - v_{m1}) & E(v_3 - v_{m3})(v_2 - v_{m2}) & E(v_3 - v_{m3})^2 \end{bmatrix} \tag{1-5}$$

with the terms $v_{mi}$ the mean value of the $i^{th}$ nodal voltage. The term $E(x)$ represents the *expected value of* $x$, given by:

$$E(x) = \frac{1}{N} \cdot \sum_{i=1}^{n} x_i \tag{1-6}$$

where the summation is over the $N$ Monte Carlo iterations.

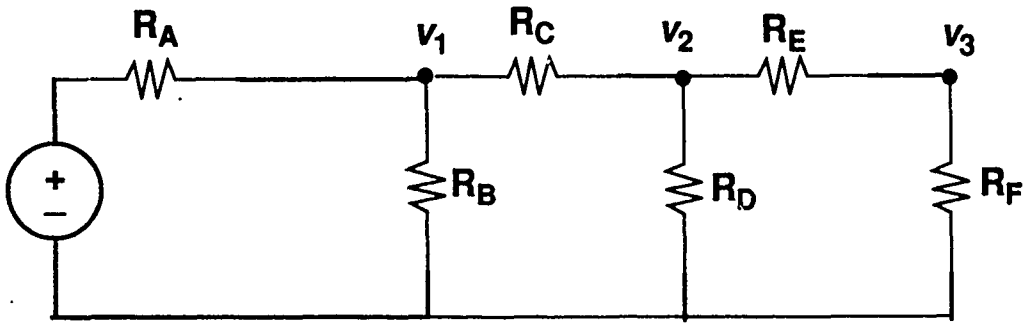



***Figure 1-5.*** Hypothetical circuit yielding multivariate statistics. The voltages $v_1$, $v_2$, and $v_3$ are the components to the multivariate vector $V$ which undergoes analysis. Components $R_A$, $R_B$, $R_C$, $R_D$, $R_E$, and $R_F$ can be nonlinear resistors.

A correspondence between the univariate and multivariate cases should now become apparent. The covariance matrix is the multivariate counterpart to the univariate standard variance (compare eqs. 1-2 and 1-5). Likewise, individual voltages are now grouped into vectors.

Note the number of components in $V$ can be augmented through means other than by simply looking at more nodal voltages. Additional measurements, such as AC magnitude and phase, frequency swept nodal voltages, or nodal voltages as a function of input voltage all contribute additional components to $V$. Each AC phasor measurement contributes two components (real and imaginary parts, or magnitude and phase). Thus, even the circuit in Figure 1-1 can be treated as a multivariate case when, for example, the input voltage source is swept over voltage and/or frequency. As will be seen in the examples, additional measurements can enhance the discrimination power of the classification. However, too many measurements can unnecessarily reduce computational efficiency and ultimately affect discrimination accuracy.

### *Quadratic Discrimination Score (or Quadratic Distance Measure)*

Multivariate discrimination requires the computation of a quantity analogous to the univariate normalized distance (eq. 1-3). This quantity, called the quadratic discrimination score, is computed by first taking the natural logarithm of both sides of eq. 1-4:

$$ln\left(f_k(V)\right) = \left(-\frac{p}{2}\right)ln(2\pi) - \frac{1}{2}ln\left|\Sigma_k\right| - \frac{1}{2}\left((V-\mu_k)^T\Sigma_k^{-1}(V-\mu_k)\right) \quad (1\text{-}7)$$

where $k$ = 1,2,3,...$g$ (for $g$ fault classes), and $p$ is the degree of freedom for the distribution, which is equivalent to the number of voltage components in the vector $V$. Given a voltage vector $V$ from the test circuit, the fault distribution that $V$ most likely belongs to corresponds to that distribution which maximizes the quantity $ln[f_k(V)]$. Eq. 1-7 is evaluated for each of the $k^{th}$ fault classes. Since we are comparing different values of number evaluated in eq. 1-7, we can eliminate the constant term $(p/2)ln(2\pi)$. With a sign change, we then have:

$$d(V)_k = ln\left|\Sigma_k\right| + \left((V-\mu_k)^T\Sigma_k^{-1}(V-\mu_k)\right) \qquad (k = 1,2,3,...g) \qquad (1\text{-}8)$$

$d_k(V)$ is the *quadratic discrimination* score. The $k^{th}$ of $g$ fault types resulting in a minimum score identifies the type of fault detected in the test circuit. Note the resemblance of $d_k(V)$ with the normalized distance value for the univariate case (eq. 1-3).

This analysis assumes that the *prior probabilities* of all fault types are equal. In other words, all fault types considered during the generation of the training set are assumed to be equally likely to occur. This assumption will lead to a conservative outcome in determining whether or not a circuit is faulted. However, the assumption can also lead to misclassification among faults, especially when a commonly occurring fault must be classified. To account for the relative frequency of faults, modify equation (1-8) by:

$$d(V)_k = ln\left|\Sigma_k\right| + \left((V-\mu_k)^T\Sigma_k^{-1}(V-\mu_k)\right) + ln(p_k) \qquad (k = 1,2,3,...g) \qquad (1\text{-}9)$$

where $p_k$ is the prior probability of the $k^{th}$ fault type. The prior probabilities among all the fault types must sum to unity. Note that this analysis also assumes that all misclassification costs are equal (see ref.16 for more details).

Although eq. 1-8 appears relatively easy to evaluate, care must be taken in computing the term $\Sigma^{-1}$. Often $\Sigma$ is singular because of the presence of identical voltage components obtained during the training process. In other words, tests or simulations that generate linearly dependent voltage components will lead to a singular covariance matrix. To avoid this situation, voltages must be carefully chosen such that they are linearly independent. Programs such as SAS also test $\Sigma$ for singularities,

and remove linearly dependent rows and columns. In certain situations, many rows and columns (hence, voltage vector components) have to removed from the covariance matrices of most of the fault classes. In such cases, the pooled covariance matrix can be used. This matrix is a weighted average of the individual fault type covariance matrices:

$$\Sigma_{pooled} = \frac{(N_1 - 1)\,\Sigma_1 + (N_2 - 1)\,\Sigma_2 + \ldots + (N_g - 1)\,\Sigma_g}{N_1 + N_2 + \ldots + N_g} \qquad (1\text{-}10)$$

where $N_k$ is the number of Monte Carlo samples for each $k^{th}$ of $g$ fault classes. When the pooled covariance is used, the user risks a potential degradation in discrimination power. In SAS, the use of the pooled covariance matrix is selected by the user. In the demonstration program of this study, a fault type will use the pooled covariance matrix when for the $k^{th}$ fault type, $ln|\Sigma_k| < -224$ (i.e., when the $k^{th}$ fault's covariance matrix determinant becomes extremely small).

### Discrimination Testing

The power of a discriminator is simply a measure of how often the discriminator calculation (eq. 1-8) correctly and incorrectly classifies a test point. In the present study, testing is performed by submitting a series of test points having known fault types to the discriminator. A summary is then tallied describing how many faults are correctly and incorrectly classified, and under what fault classes misclassified faults are assigned. This summary can be presented in the form of a *confusion table*, shown in Figure 1-6 for a discrimination analysis in which three fault types are considered. With $N_1$, $N_2$, and $N_3$ the number of tests for each fault type, and the subscripts $m$ and $c$ meaning misclassified and correctly classified, respectively, we define the *apparent error rate of misclassification* (APER) as ref. 16; pp. 496-497]:

$$APER = \frac{N_{1m} + N_{2m} + \ldots + N_{gm}}{N_1 + N_2 + \ldots + N_g} \qquad (1\text{-}11)$$

Discrimination analysis summaries are provided by SAS and the demonstration program (examples are given in Part 2).

| | | *Predicted Membership* | | |
|---|---|---|---|---|
| | | *class 1* | *class 2* | *class 3* |
| *Actual Membership* | *class 1* | $N_{1c}$ | $N_{12m}$ | $N_{13m}$ |
| | *class 2* | $N_{21m}$ | $N_{2c}$ | $N_{23m}$ |
| | *class 3* | $N_{31m}$ | $N_{32m}$ | $N_{3c}$ |

***Figure 1-6.*** Confusion table for three classes. The symbol $N_{ijm}$ is the number of actual elements in the $i^{th}$ class misclassified as belonging to the $j^{th}$ class; $N_{ic}$ is the number of elements in the $i^{th}$ class correctly classified as belonging to that class. The total misclassification count for the $i^{th}$ class, $N_{im}$, is given as: $N_{im} = \sum_{j,j \neq i} N_{ijm}$

## 1.2C. Overall Procedure

We now summarize the overall procedure used in this study to evaluate the use of discrimination analysis in classifying faults in analog circuits.

*Step 1:* Assign normal distributions to all circuit components that affect the measured performance of a given circuit

*Step 2:* Perform a series of Monte Carlo simulations of the circuit operating under nominal and faulted conditions. For simplicity, consider only one catastrophic fault at a time. Store selected node voltages from each simulation run in a file. When this step is completed, the training set has been generated.

*Step 3:* Perform a new series of Monte Carlo simulations on the same circuit, again under nominal and faulted conditions. For each simulation run, evaluate the quadratic discrimination score and assign the circuit fault to the appropriate fault class. Tally how often a given fault is correctly classified. Also keep track of what fault classes a given fault is incorrectly classified under.

One means of saving time and computer disk space is to simply use the training set data as the test data of Step 3. This "trick" eliminates the need to perform a new series of Monte Carlo simulations. The drawback is that when test and training set data are the same, the APER becomes underestimated. However, this problem becomes insignificant for sample sizes over 50 for each fault type, as will be seen in Part 2.

The above steps are carried out by the Statistical Fault Analyzer (SFA) program written for this study, in conjunction with the commercial statistical package SAS (and many other commercial packages). Details of the SFA program will be given in Section 1-4.

## 1.3. Go/No-Go Testing

Our approach to Go/No-Go testing makes use of *hypothesis testing*. This procedure is much simpler than the discrimination analysis because Monte Carlo simulations of only the nominal circuit are used to construct a training set. The training set is used to define the multidimensional probability density distribution (eq. 1-4) of the nominal circuit voltages. During hypothesis testing, we assess the likelihood that a voltage test vector $V$ falls within the distribution.

Assuming that $V$ takes on a multivariate normal distribution (which is usually reasonable for the purposes of our study), it can be shown that the solid ellipsoid of $V$ vectors satisfying the relation [ref. 16, p. 126]:

$$(V-\mu_{nominal})^T\Sigma^{-1}_{nominal}(V-\mu_{nominal}) \le \chi^2_p \qquad (1\text{-}12)$$

has probability 1 - $\alpha$. Here $X_p^2(\alpha)$ is the $100\alpha^{th}$ percentile of a chi-square distribution with $p$ degrees of freedom (in effect, $p$ is the number of voltage components in $V$); $\mu_{nominal}$ is the vector containing the average of the voltage components obtained from measurements or simulations of the nominal circuit, and $\Sigma_{nominal}$ is the nominal circuit's covariance matrix.

A hypothesis test is performed by evaluating eq. 1-12 for a given test circuit's $V$. When the result is less than $X_p^2$, the test circuit's performance has a 100(1-$\alpha$)% chance of being nominal. If eq. 1-12 results in a value greater than $X_p^2$, the circuit performance does not resemble that of a nominal circuit; hence, the circuit fails the Go/No-Go test. In terms of traditional hypothesis testing, we

define the null hypothesis as "the circuit is nominal" (i.e., good). The null hypothesis is rejected when eq. 1-12 results in a value greater than $X_p^2$; in this case the circuit fails. Note that the probability of falsely rejecting the null hypothesis is $100\alpha\%$, which corresponds to a Type I error. For this study, $\alpha$ has been set to 0.05.

As in the discrimination analysis, care must be taken when evaluating eq. 1-12 due to the possibility that $\Sigma$ is a singular matrix. If $\Sigma$ is singular, rows and columns must be deleted that correspond to linearly dependent voltage components.

We note that an "unofficial" method for performing hypothesis testing makes use of the discrimination analysis. This is performed by generating a training set for the nominal circuit, then lumping all of the training sets for the faulted circuits into one fault class. During hypothesis testing, a given test circuit will either be categorized in the "nominal" or "faulted" class (or "none" upon setting the distance threshold to a nonzero value). Although this method often results in less error than the use of eq. 1-12, it is potentially dangerous because the centroid (i.e., mean) of the faulted voltage vector distribution can lie directly on the nominal circuit's centroid. Likewise, the discrimination analysis would only account for faults included in the training set. In contrast, eq. 1-12 works for any fault whose voltage vectors are different from the nominal $V$.

Part 2 illustrates several examples of the use of hypothesis testing.

## 1.4. Statistical Fault Analyzer (SFA)

The classification and Go/No-Go concepts discussed in Sections 1.2 and 1.3 have been implemented in the SFA program that accompanies this report. Figure 1-7a describes the flow of the program when operating in the training set generation mode. The SFA was written in C and intended for operation under UNIX on SUN Microsystems series 3, 4, or SPARCstation workstations. We now discuss the program flow.

### *Input File*

Circuits are described in an ASCII file containing modified SPICE syntax. Figure 1-8 shows a typical file for a one-stage amplifier. The file completely follows SPICE conventions, except for add-

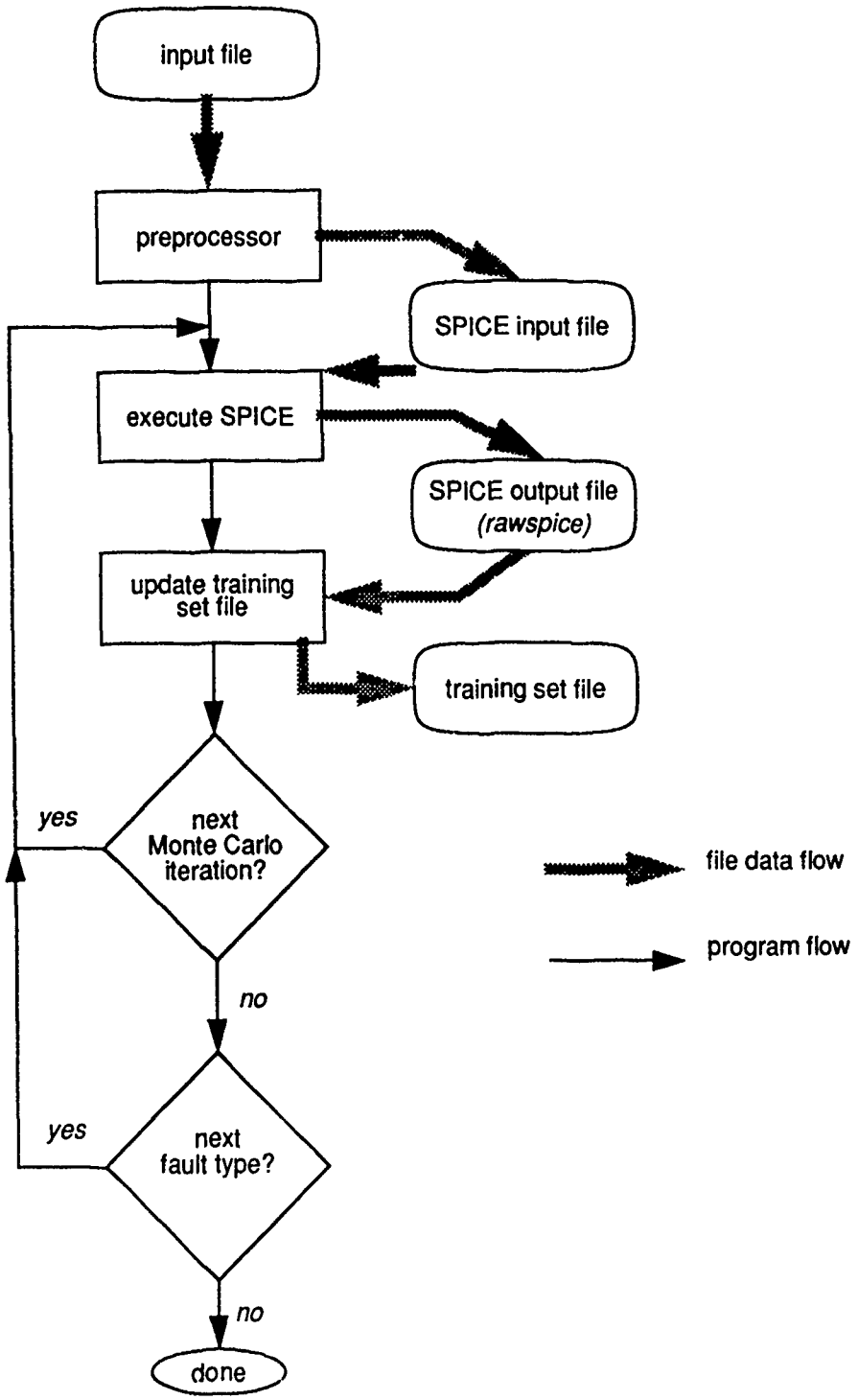



***Figure 1.7A.*** Overall flow of SFA program for the generation of training set data and testing by SAS.

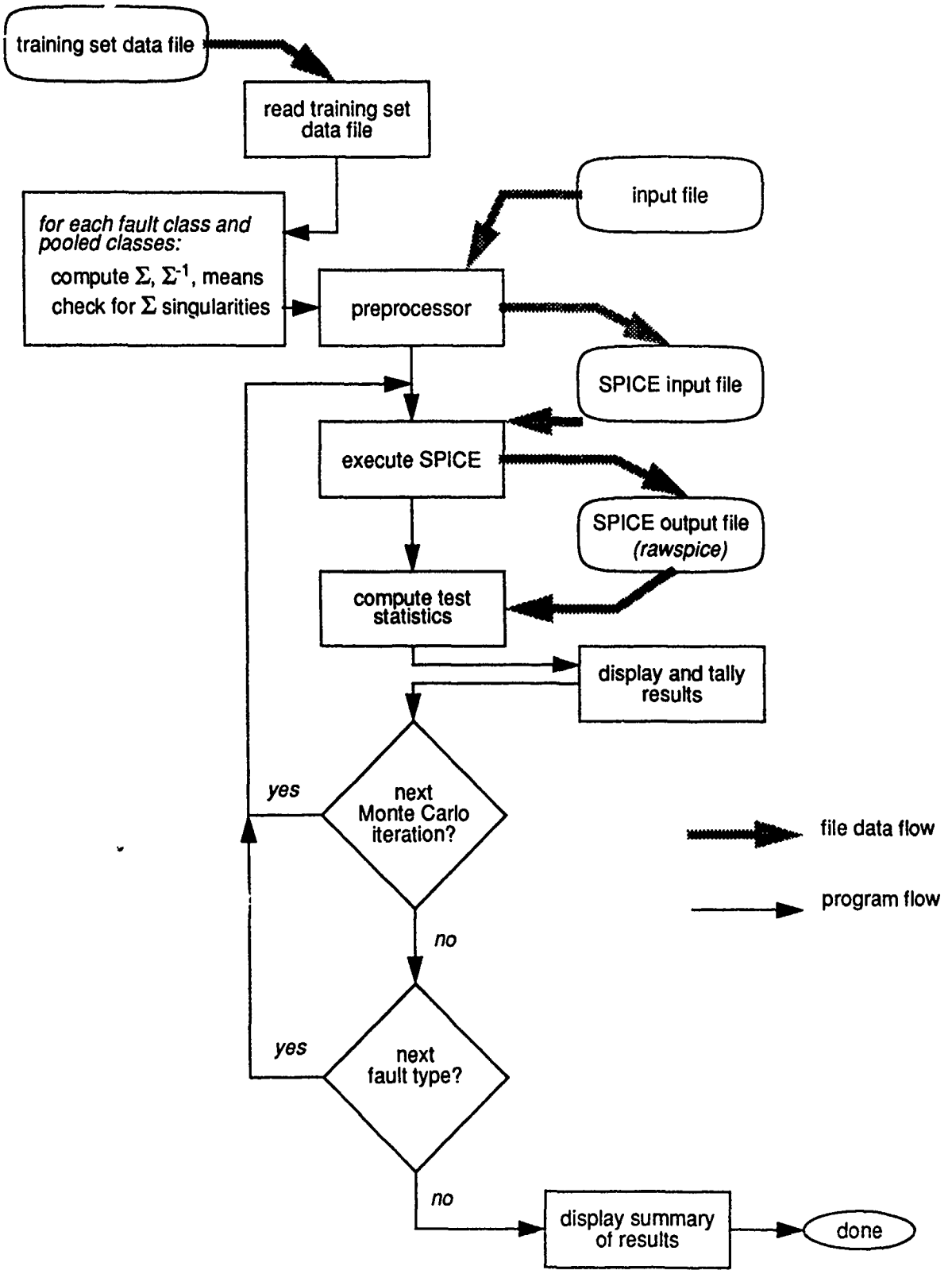



***Figure 1.7B.*** Overall flow of SFA program for the testing of training set data.

ed syntax that controls the SFA. Lines beginning with the symbol # 1) identify a faulted circuit component by name, 2) the faulted component's electrical value when faulted, 3) how many Monte Carlo iterations are to take place for the given fault, 4) an identification label for the given fault (to define the name of the fault class in SAS), and 5) when the faulty component is represented by a SPICE device model, the name of the model parameter. One special line, labeled **#GOOD N** tells the SFA to simulate the nominal (i.e., good) circuit N times. Lines beginning with **%dc** are followed by node numbers. These numbers tell the SFA to dump the DC voltages of circuit nodes having these numbers to the training set file. Likewise, the **%ac** lines dump the indicated AC nodal voltages to the training set file. By default, real and imaginary parts are stored in the training set file; however, the key word **%POLAR** will cause AC voltages to be stored as magnitude and phase.

Pairs of quantities in square brackets define the normal distributions assigned to a given component. The first value is the nominal value for a component; the second value represents a standard deviation of one sigma. When a component is faulted, the square brackets for that component and the values within them are replaced by the fault value specified in the # line for that component. For example, the fault type labeled RE_OPEN would correspond to 3 simulations with the resistance value of RE set at 1 megΩ. After each of the 3 simulations, the DC nodal voltages at nodes 1, 2, 3, and 7 would be stored in the training set file, and identified by the label RE_OPEN. Likewise, the magnitude and phase of the voltages at nodes 2, 3, and 7 would also be stored in the training set file.

### *Preprocessor*

The preprocessor in the SFA reads in the input file, interprets the special input file symbols described above, and outputs a standard SPICE file. The SPICE file can be viewed by the user by introducing the key word **%TRACE** into the input file.

### *SPICE Execution*

Once the SPICE file is generated, it is sent to a batch version of SPICE3C1 (the program is named *bspice*, and is available from the Univ. of CA, Berkeley). bspice is executed without operator in-

```
Single-stage amplifier
#GOOD 3
#RC 1.0e6 2 RC_OPEN
#RE 1.0e6 3 RE_OPEN
#RB 1.0e6 2 RB_OPEN
#RB 1.0 2 RB_Short
#Rbc 1.0 4 RBC_Short
#Rbe 1.0 2 RBE_Short
#Rce 1.0 2 RCE_Short
#QNL 1.0 2 Q_low BF
#QNL 200.0 2 Q_hi BF
%dc 1 2 3 7
%ac 2 3 7
%POLAR
VCC 8 0 5.0
RSDC 8 2 1.0
VIN 10 0 AC 1
CIN 10 1 100.0UF
R3 2 3 [1.2,.06]K
RC 3 6 [1.0,0.0]
RB 1 5 [100.0,5.0]
R2 2 1 [61.0,3.1]K
R1 1 0 [17.6,0.9]K
RE 4 7 [1.0,0.0]
R4 7 0 [300.0,15.0]
Rbc 5 6 [1.0e8,0.0]
Rbe 5 4 [1.0e8,0.0]
Rce 6 4 [1.0e8,0.0]
Q1 6 5 4 QNL
.op
.AC LIN 1 100 100hz
.model QNL NPN(BF=[80,12] CCS=2PF TF=0.3NS TR=6NS CJE=3PF
+ CJC=2PF VA=[50,7.5])
.END
```

***Figure 1-8.*** SFA input file for single-stage amplifier. See Figure 2-1A for circuit diagram.

tervention by UNIX system shell routines invoked in the SFA code. NOTE: The version of SPICE3C1, as supplied by UC Berkeley, contains several bugs that have been corrected for operation with the SFA. All SPICE3C1 bugs, however, have not been corrected.

### SPICE Output

Output from bspice is sent to a file called *rawspice*. SFA scans this file to extract the nodal voltages specified in the **%ac** and **%dc** lines of the input file. The user does not ever work with the *rawspice* file. Any errors encountered during SPICE execution are reported in a separate output file.

### Training Set Data

With proper operation of the SFA, a new file is generated containing the training set data. Figure 1-9 shows the training set file resulting from the input file of Figure 1-8. Note that to keep the size of Figure 1-9 within reason, the number of Monte Carlo simulations specified in Figure 1-8 for each fault type is extremely low. Normally this number would be well over 100. When the generation of the training set data us completed, the resulting file is used by SAS or the SFA (in the test mode) to test the discrimination.

#### *Discrimination Testing and Hypothesis Tests*

Discrimination testing calls for a slightly different operation of the SFA, as shown in the flow diagram of Figure 1-7b. Now the training set data generated in the steps above is read into the program and used to compute the quadratic distance and hypothesis value for newly generated test data. The results are displayed as each test point is classified and given a Go/No-Go test.

The discrimination analysis can alternatively be tested using the SAS DISCRIM routine. The SFA generates data files whose format is compatible with input file descriptions defined in a SAS control file. An example SAS control file is shown in Table 2-1B. Refer to the *SAS Users Guide on Statistics* [13] for a description of this file.

```
2 3 4 5 6 7 8 9 10
NOMINAL 4.999e+00 1.024e+00 3.891e+00 2.535e-01 -3.030e-03 -5.033e-06
 -3.933e+00 -6.533e-03 8.995e-01 1.497e-03
NOMINAL 4.999e+00 1.045e+00 3.933e+00 2.731e-01 -2.926e-03 -4.926e-06
 -3.535e+00 -5.951e-03 9.052e-01 1.527e-03
NOMINAL 4.999e+00 9.579e-01 4.231e+00 1.954e-01 -2.827e-03 -5.053e-06
 -3.442e+00 -6.152e-03 8.748e-01 1.566e-03
RC_OPEN 5.000e+00 7.161e-01 4.994e+00 1.012e-02 1.743e-05 2.559e-07
 2.485e-04 3.653e-06 2.447e-01 3.594e-03
RC_OPEN 5.000e+00 7.156e-01 4.994e+00 1.093e-02 1.627e-05 2.252e-07
 2.619e-04 3.627e-06 2.589e-01 3.583e-03
RE_OPEN 5.000e+00 1.094e+00 4.999e+00 1.586e-04 1.480e-05 1.777e-08
 -1.154e-03 -2.750e-07 2.887e-04 3.328e-07
RE_OPEN 5.000e+00 1.032e+00 4.999e+00 1.381e-04 1.513e-05 1.958e-08
 -1.096e-03 -3.368e-07 2.809e-04 3.508e-07
RE_OPEN 5.000e+00 1.123e+00 4.999e+00 1.577e-04 1.515e-05 1.810e-08
 -1.237e-03 -2.974e-07 2.734e-04 3.138e-07
RB_OPEN 5.000e+00 1.166e+00 4.949e+00 1.243e-02 -6.149e-05 -5.152e-08
 -9.383e-02 -8.585e-05 2.304e-02 2.120e-05
RB_OPEN 5.000e+00 1.304e+00 4.924e+00 1.961e-02 -7.805e-05 -6.435e-08
 -1.143e-01 -1.014e-04 2.941e-02 2.623e-05
RB_SHORT 4.999e+00 1.062e+00 3.797e+00 2.896e-01 -2.966e-03 -4.601e-06
 -3.790e+00 -5.879e-03 9.128e-01 1.419e-03
RB_SHORT 4.999e+00 9.848e-01 4.114e+00 2.197e-01 -2.924e-03 -5.037e-06
 -3.589e+00 -6.184e-03 8.900e-01 1.536e-03
RBC_SHORT 4.997e+00 1.600e+00 1.606e+00 7.999e-01 5.893e-04 2.799e-05
 7.005e-01 3.327e-02 6.796e-01 3.228e-02
RBC_SHORT 4.997e+00 1.619e+00 1.626e+00 8.206e-01 5.998e-04 2.826e-05
 7.066e-01 3.329e-02 6.861e-01 3.233e-02
RBC_SHORT 4.997e+00 1.696e+00 1.705e+00 8.961e-01 6.601e-04 3.145e-05
 6.949e-01 3.311e-02 6.766e-01 3.224e-02
RBC_SHORT 4.997e+00 1.585e+00 1.592e+00 7.856e-01 5.901e-04 2.854e-05
 6.955e-01 3.363e-02 6.744e-01 3.261e-02
RBE_SHORT 5.000e+00 3.153e-02 5.000e+00 2.327e-02 1.699e-05 7.304e-07
 3.370e-05 1.988e-06 7.366e-01 3.164e-02
RBE_SHORT 5.000e+00 3.510e-02 5.000e+00 2.649e-02 1.762e-05 7.057e-07
 3.743e-05 2.138e-06 7.534e-01 3.015e-02
RCE_SHORT 4.996e+00 1.222e+00 1.080e+00 1.070e+00 1.694e-05 1.942e-08
 8.483e-06 8.229e-07 8.430e-06 8.203e-07
RCE_SHORT 4.996e+00 1.140e+00 1.038e+00 1.028e+00 1.623e-05 1.907e-08
 8.085e-06 7.880e-07 8.033e-06 7.854e-07
Q_LOW 5.000e+00 7.056e-01 4.964e+00 1.880e-02 -6.748e-04 -7.606e-06
 -7.607e- 1 -8.574e-03 3.944e-01 4.447e-03
Q_LOW 5.000e+00 7.092e-01 4.956e+00 2.064e-02 -6.771e-04 -7.775e-06
 -8.753e-01 -1.005e-02 4.143e-01 4.759e-03
Q_HI 4.999e+00 1.048e+00 3.920e+00 2.761e-01 -2.967e-03 -4.085e-06
 -3.553e+00 -4.892e-03 9.083e-01 1.253e-03
Q_HI 4.999e+00 1.011e+00 4.111e+00 2.432e-01 -2.887e-03 -4.049e-06
 -3.283e+00 -4.604e-03 8.983e-01 1.262e-03
END
```

***Figure 1-9.*** Training set file generated by SFA and readable by SAS.

The examples provided in Part 2 provide plenty of illustrations of SFA and SAS output. Study of these examples will make the operating details of the SFA more apparent. Further information on the use of the SFA program is given in Appendix A.

## 1.5. Other Details

### 1.5A. Random Numbers

The Monte Carlo simulations called for random variations in key circuit component values. These variations take on normal distributions with user-specified means and standard deviations. SFA generates normal distributions based upon the central limit theorem by the simple C routine:

```
float nran2(mean,sigma) /* generates normal distribution of mean */
float mean,sigma; /* "mean" and sigma "sigma" */
 /* note: sigma corresponds to 1 sigma deviation */
{
 int i;
 float u;

 u =0.0;
 for(i=0;i<12;++i)u += ran2(&iseed);
 return (sigma*(u-6.0) + mean);
}
```

The function ran2 supplies uniformly distributed random numbers between 0 and 1 (but not inclusive). Within ran2, pseudo random numbers are generated by the *linear congruential method*[9], then sent to a shuffling routine to ensure random uniformity. The large modulus seed (a value of 714025) of the generator is sufficiently large to avoid any chance of random number cycling during the simulations. Furthermore, a 100-bin Chi-squared test applied to ran2 indicates that, to a 5% level of confidence, the function's output is random.

## 1.5B Plackett-Burman Experimental Designs

In complex simulations, there are many model parameters that can be modified. However, some of these parameters will have little or no effect on the behavior of the simulation, while a relatively few may almost completely determine the simulation output. When all possible parameters are varied simultaneously, the number of simulation runs required grows exponentially to an impractical number. If it is the case that the space of simulation outputs can be "spanned" by varying only a few parameters, then the number of simulations can be reduced tremendously with

no loss of confidence in the results. In the circuit simulations of this study, many Monte Carlo iterations often appear necessary to account for the variability of the circuit component parameters. Parameter screening can reduce the number of simulations while maintaining a good representation of circuit variability.

One approach to determining a smaller subset of parameters is by designing experiments for the simulation. Each parameter under consideration is varied in a specific way, determined by the particular experimental design used, and the resulting output of the simulation is recorded. Experimental designs can become complex, and can be used to estimate not only the first-order main effects of each parameter, but also second-level effects and interactions between parameters. However, for our purposes, we are interested only in determining which model parameters have some effect and should be included in any further simulation studies, and which parameters have little effect and can be safely ignored.

To demonstrate this procedure to reduce the number of simulation model parameters, we used a *Plackett-Burman experimental design* [12]. These designs are two-level, main-effect-only fractional factorial designs which allow for the efficient estimation of main-factor effects. Although relatively simple, they are effective for screening out a large number of factors before a more complete experimental or response surface design is implemented. One caveat in their use is that the main effects cannot be estimated in an unbiased manner unless 1) all interactions between pairs of factors is negligible; or 2) there are only a few important factors.

Section 2-1K discusses in detail the application of Plackett-Burman experimental design to a one-stage amplifier.

## 1.5C. Stepwise Discrimination Analysis

The objective of *stepwise discrimination* is to determine the subset of variables which can best be used to discriminate classes in a discrimination analysis. If too many variables (e.g., voltages) are included, many of them might be highly correlated, the resulting variance-covariance matrices will be singular or ill-conditioned, and the power of the discrimination will be reduced due to numerical instability. If too few or incorrect variables are selected, then discrimination analysis will give incorrect results. With stepwise discrimination, we screen the nodal voltages to arrive at an efficient

fault classifier. This contrasts to the Plackett-Burman screening discussed above, where simulation parameters that contribute minimally to circuit variability are screened out of the analysis.

Stepwise discrimination functions similarly to stepwise regression in that a sequence of additions and deletions to the set of variables included in the discrimination analysis is undertaken. The decision to add or remove a variable is based on a statistical measure, called *Wilk's lambda*, which evaluates the increase or decrease in the discriminatory power of the classification.

Wilk's lambda provides a measure of the scatter within a set of classes compared to the total scatter among all classes. It is the multi-dimensional equivalent to $\eta^2$ (*Fisher's Correlation Ratio*) in analysis of variance, where:

$$\eta^2 = \frac{SS_w}{SS_G} \tag{1-13}$$

Here $SS_w$ is the average within-class sum-of-squares over all classes, while $SS_G$ is the global sum-of-squares computed over all data points relative to the global mean $V$. To obtain the multi-dimensional equivalent using the vector $V$ of voltage variables $v$, first define $T(V)$ as the variance-covariance matrix for all data together, and $W_k(V)$ as the variance-covariance matrix for the $k^{th}$ class of $g$ classes. Then, with:

$$W = \sum_{j=1}^{g} W_k(V) \tag{1-14}$$

*Wilk's Lambda* becomes defined as:

$$\Lambda = \frac{|W(V)|}{|T(V)|} \tag{1-15}$$

which has a value between 0 and 1. Large values indicate poor separation among the groups, while low values indicate good separation. In order to determine the effect of adding a voltage variable, $u$, to the existing set of variables in $V$, we define the *partial lambda-statistic* as:

$$\Lambda(uV) = \frac{\Lambda(V,u)}{\Lambda(V)} \tag{1-16}$$

(Note, $uV$ is not the multiplication of $V$ by the scalar $u$). The corresponding F-statistic is:

$$F = \frac{n-g-p}{g-1} \cdot \frac{(1-\Lambda(uV))}{\Lambda(uV)} \tag{1-17}$$

where $n$ is the number of observations, $p$ is the number of variables in $V$, $g$ is the number of classes,

and $u$ is used to test the significance of the change in $\Lambda(V)$ resulting from the addition of the voltage variable $u$. This statistic is the *F-to-enter statistic* for the entry of $u$ into the set $V$, or the *F-to-remove statistic* for the deletion of $u$ from the set $(V,u)$. Determination of the entry or exit of the variable $u$ depends on whether this F-statistic is greater or less than a given threshold which is set by the user. At each step, the F statistic is computed for each of the voltage variables in $V$. The variable whose F-statistic is least *and* less than the tolerance level is removed. This removes the variable which reduces the discriminatory power of the analysis the least. If no variable is removed, then all variables not included in $V$ are examined, and the one for which the F-statistic is largest and which satisfies the criterion to enter is added. This continues until no further changes to the variable set occur. Stepwise discrimination, as used in this study, begins with no variables included in $V$.

While an automatic variable selection technique may not provide the best model for a specific data set, its cautious application can provide considerable information. For best results, the output of a stepwise discrimination analysis should be used as a supplement to the user's knowledge of the data.

### 1.5D. Nonparametric Discrimination

In some situations, the inherent assumption of normality in the distributions of the simulation output voltage vectors $V$ may not hold. In this case, discrimination analysis must resort to a *nonparametric method.* One simple technique explored in this study makes use of the "nearest k neighbors" method. Figure 1-10 illustrates this method for a simple bivariate classification. In this case, the voltage vectors are made up of the two voltage components $v_1$ and $v_2$. Three types of faults are possible in the example, and our aim is to determine which of the three faults the test point **x** belongs. The training set data for the three fault types is shown in Figure 1-10, where each fault type takes on one of three types of points (a square, triangle, or circle). The nearest k neighbors method simply tallies how many training set points from each fault type fall near the test point. Only the k nearest neighbors to the test point are included in the tally (hence the name of the method).

A major drawback of this technique is that it potentially requires long computational time since the distances between the test point and all points in the training set must be computed. The distances must then be sorted. In view of the long computations, this technique was used sparingly, and yielded similar results to the classical discrimination method when it was used. Preliminary screening of training set data points can reduce the computation time of determining the distances.

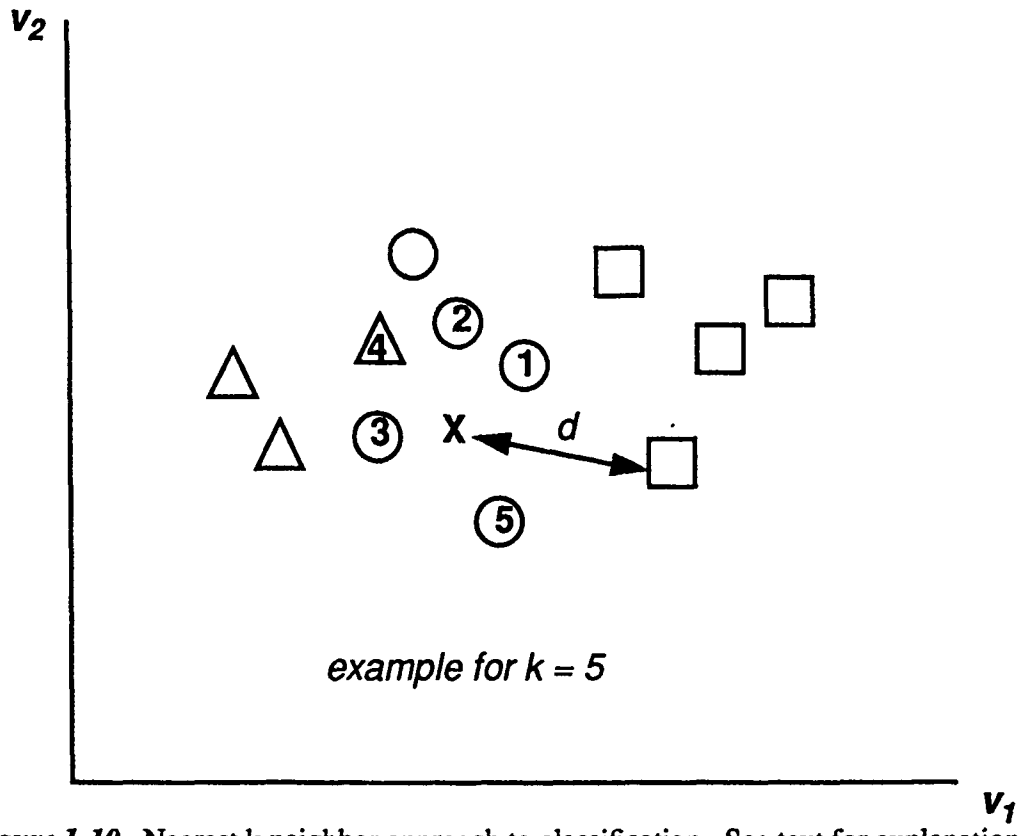



***Figure 1-10.*** Nearest k neighbor approach to classification. See text for explanation.

## 1.6. Direct Circuit Solution

Earlier in the program, circuit simulations were carried out by directly solving the systems of equations that describe the circuits. Both DC and AC solutions were carried out. The nonlinear equation solving routine HYBRD was used to determine the DC operating points, while LINPACK routines solved the system of AC phasor equations to determine the AC response of the circuit. Data resulting from the direct solutions were dumped to ASCII files, and subsequently analyzed using SAS in a manner similar to that described for the SFA software. Direct solution was later abandoned in favor of the SPICE-based approach that was employed throughout most of the program. Nevertheless, the direct solution method helped to provide verification of the SPICE solutions during the transition to SPICE.

## 1.7. Modeling Error Considerations

### 1.7A. Introduction

All of the analyses described in Part 2 of this report have relied on simulations and the credibility of the SPICE device models to perform fault classifications, and to determine "nominal" operating conditions for Go/No-Go testing. Simulations were required because of the obvious impracticality of fabricating hundreds or thousands of circuits for statistically describing the distributions of even the major fault categories only. Given the probability of encountering a specific fault type, thousands of circuits may be required to observe only 100 circuits of the given fault type. Moreover, extensive measurements and physical inspection of the faulted circuits would be required to ascertain the fault type present. Thus, the fabrication of ICs for the purpose of building a statistical database is impractical, except when IC fab lines having high production volume are available as a source of the IC data. Simulation and modeling provide a practical alternative to device fabrication and inspection for faults.

The fault detection and classification algorithms that we have used make use of simulated data, not real measurement data. Therefore, how can the classification and fault detection algorithms derived from the simulated data be used in practice to predict faults in real linear microcircuits? Below we discuss a method for verifying and calibrating the simulations in order that the fault discrimination rules will apply to actual circuit data.

### 1.7B. Verification and Calibration Method

Model verification and calibration begins by obtaining real measurement data from samples of ICs. By simply comparing the response of a given circuit to its specification, a determination can be made whether or not the circuit is acceptable; separation of good and bad circuits then follows, with further separation by fault type when possible. Collecting the responses observed from many nominal and faulted circuits enables one to statistically describe the response of these circuits. Using the statistical method of hypothesis testing, the distribution of the actual data can then be compared to the distribution obtained from runs made using the simulation model where the circuit parameters are randomly varied over their specification ranges, with and without the presence of faults. A determination of whether or not the model adequately describes the actual circuit behavior is then

made. If the model does not adequately describe the actual circuit behavior, then the model is corrected and/or calibrated so that it does. Once these model adjustments are made so that the data from the simulation model and the data collected on actual circuits are statistically indistinguishable, the simulation model can be used to construct the fault discrimination rules for classification.

We now describe an algorithm for model verification and calibration:

*Step 1:* Collect output response data from actual circuits taken from a cross section of production lots, wafer location, manufacturers, etc. in order to represent the variation in the circuit's response that can be expected during manufacture or in the field. Segregate the nominal (or acceptable) circuits from the faulted (unacceptable) circuits using the criterion of whether or not the output responses are within the specification limits. For statistical reasons, the number of nominal circuits should exceed 30 + (the number of responses). For example, if there are 5 output responses (e.g., 5 voltage measurements), output response data should be collected from at least 35 nominal circuits.

*Step 2:* Statistically test whether or not the simulated circuit data are equivalent to the actual circuit data. Bin the actual data into $K$ bins. If the number of different types of output responses N exceeds 1, then these bins represent regions in N-space. Count the number of observations from both the actual ($a_i$)and simulated ($s_i$) data that lie in each of these bins. The binning of the data should be such that the bins cover the entire data set and a reasonable number of actual circuit responses all lie within each bin. Then perform a statistical Goodness-of-Fit test to determine whether or not the two distributions are statistically indistinguishable. For example, compute:

$$\chi^2 = \sum_{i=1}^{K} \frac{(a_i - s_i)^2}{a_i} \tag{1-17}$$

where $a_i$ and $s_i$ are the number of actual and simulated circuit output responses falling in the $i^{th}$ region (or bin), respectively. If $X^2$ is greater than the critical Chi-square value with $K$-1 degrees of freedom and a significance level taken from the standard statistical Chi-square tables, then reject the hypothesis that the simulated

data and the actual data are equivalent and go to Step 3. Otherwise, accept the hypothesis that the two distributions are indistinguishable and stop.

*Step 3:* Correct any modeling errors, or errors in assumptions, and go back to Step 1. If there are no apparent modeling errors, then calibrate the model to the actual data by adjusting the mean and/or rescaling the data for better match of the simulated and actual data. Use this same calibration procedure to adjust the output responses from each of the fault classes as well as the nominal circuit class during the construction of the fault discrimination rules.

The goal of the above algorithm is to 1) identify major errors in either the model or the assumptions that the model is based upon, and 2) to calibrate the model to resemble actual data. The latter goal ensures that the discrimination rules generated from the simulations can be applied to real circuits. As with all models, one must sometimes rely heavily on the modeler's knowledge and/or intuition when extending the model to circuit operating conditions (e.g., faults) that cannot be empirically verified. For example, we have assumed that the SPICE device models remain valid during nominal and faulted circuit operations. Because of the impracticality of collecting data from circuits that are faulted and where the faults are known with certainty, this "leap of faith" is necessary. However, by relying on the modeler's experience and sound circuit design practices, the trends in a given circuit's response in the presence of faults often is predictable. Thus, the simulated faults should have some resemblance to reality.

# Part 2. Fault Analysis Examples

## 2.0. Introduction

In Part 2 we apply the methods described in Part 1 to various types of analog circuits. We start with the simple amplifier circuit, and analyze it through direct solution. All subsequent circuits make use of the SFA program.

## 2.1. Amplifier circuit

### 2.1A. Circuit Description

Our fault analysis study began with a simple one-stage amplifier circuit. This circuit is an NPN bipolar transistor (BJT) connected in a common emitter configuration (Figure 2-1A). The intrinsic device model (indicated by the dashed lines) follows the Gummel-Poon model in SPICE [4]. The resistors $R_{bc}$, $R_{ce}$, and $R_{be}$ have been added to emulate inter-port leakages and shorts. Nominal values for the resistors are indicated in Figure 2-1A. The transistor's nominal characteristics are those specified for a device in the 741 Op Amp [4].

The nodal DC voltages are determined by finding the solution (i.e., the nodal voltages) to the following set of equations (refer to Figure 2-1A for the node numbers):

node 1: $0 = V_1 \cdot (1/R_1 + 1/R_2 + 1/R_b) - V_2/R_2 - V_5/R_b$

node 2: $0 = -V_1/R_2 + V_2 \cdot (1/R_2 + 1/R_3 + 1/R_s) - V_3/R_3 - V_{supply}/R_s$

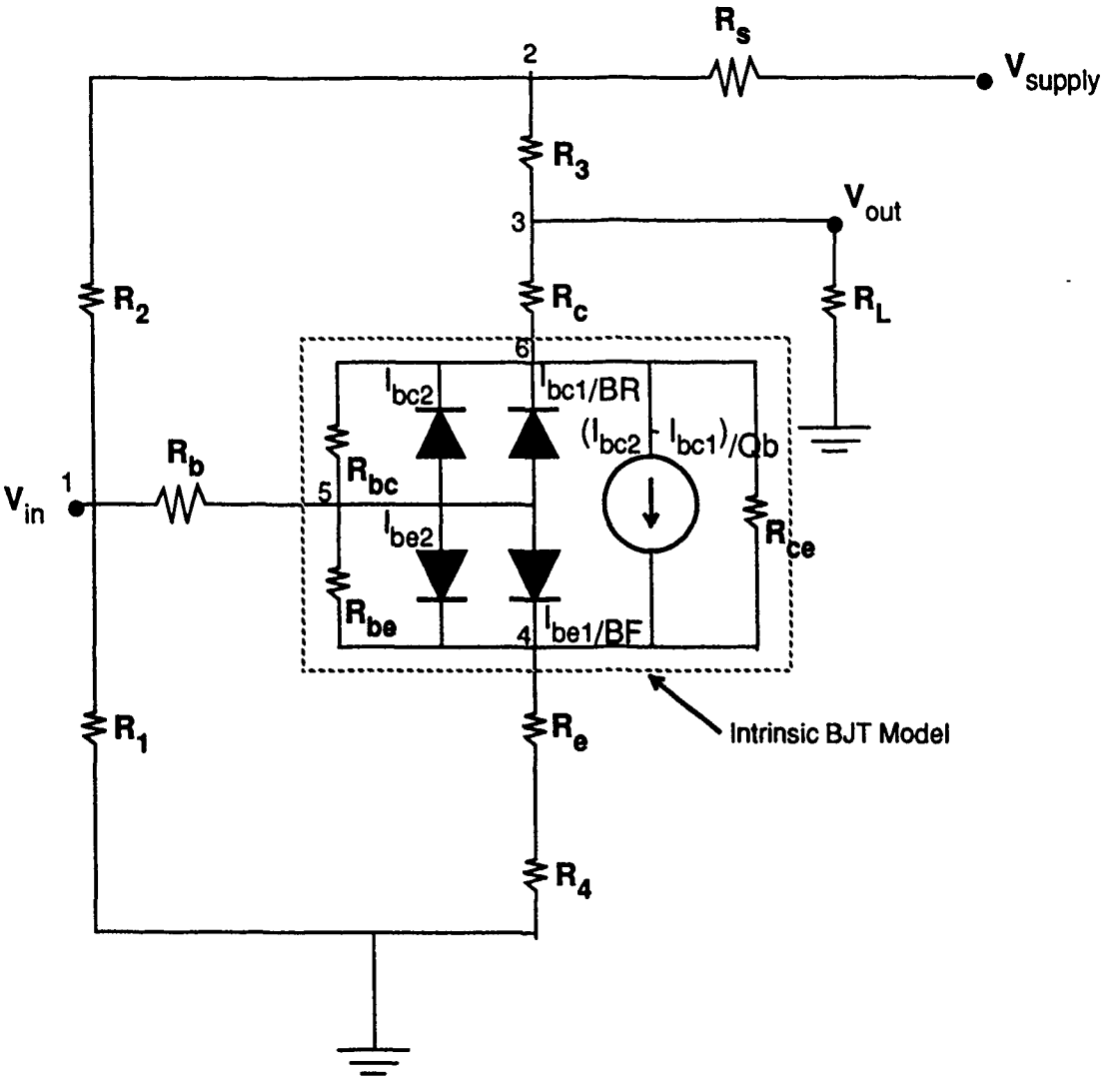
node 3: $0 = -V_2/R_3 - V_6/R_c + V_3 \cdot (1/R_3 + 1/R_c + 1/R_L)$

node 4: $0 = V_4/(R_e + R_5) + I_e$

node 5: $0 = -V_1/R_b + V_5/R_b + I_b$

node 6: $0 = -V_3/R_c + V_6/R_c + I_c$

$I_b$, $I_c$, and $I_e$ are nonlinear currents entering the base, collector, and emitter of the device, respectively; the above system of six equations in six unknowns ($V_1$ to $V_6$) is consequently nonlinear and requires special solution techniques. In our case, we have used the public-domain MINPACK HYBRD routines [8].




| | |
|---|---|
| $R_1$ | 17.6K |
| $R_2$ | 61.0K |
| $R_3$ | 1.2K |
| $R_4$ | 300.0 |
| $R_b$, $R_c$, $R_e$, $R_s$ | ~0 |
| $R_{be}$, $R_{bc}$, $R_{ce}$, $R_L$ | infinity |

***Figure 2-1A.*** Common-emitter circuit.

The device equations are those used in the Gummel-Poon model. We have followed the SPICE implementation:

base: $I_b = I_{be1}/BF + I_{be2} + I_{bc1}/BR + I_{bc2} + V_{bc}/R_{bc} + V_{be}/R_{be}$

collector: $I_c = I_{be1}/Q_b - I_{bc1}/Q_b - I_{bc1}/BR - I_{bc2} + (V_{be}-V_{bc})/R_{ce}$

emitter: $I_e = -(I_b + I_c)$

with:

$I_{be1} = IS \cdot (\exp(V_{be}/NF \cdot V_t) - 1)$
$I_{be2} = ISE \cdot (\exp(V_{be}/NE \cdot V_t) - 1)$
$I_{bc1} = IS \cdot (\exp(V_{bc}/NR \cdot V_t) - 1)$
$I_{bc2} = ISC \cdot (\exp(V_{bc}/NC \cdot V_t) - 1)$

and:

$Q_1 = 1/(1 - V_{bc}/VAF - V_{be}/VAR)$
$Q_2 = I_{be1}/IKF + I_{bc1}/IKR$
$Q_b = Q_1 \cdot (1 + SQRT(1 + 4Q_2))/2$

The parameters are as follows:

| | |
|---|---|
| BF | ideal maximum forward beta (= 80.0) |
| BR | ideal maximum reverse beta (= 1.0) |
| IS | transport saturation current (= 1.0e-16) |
| IKF | corner for forward beta high current roll-off (= 1.0e20) |
| IKR | corner for reverse beta high current roll-off (= 1.0e20) |
| ISC | base-collector leakage (= 0.0) |
| ISE | base-emitter leakage (= 0.0) |
| NC | base-collector leakage emission coefficient (= 1.0) |
| NE | base-emitter leakage emission coefficient (= 1.5) |
| NF | forward current emission coefficient (= 1.0) |
| NR | reverse current emission coefficient (= 1.0) |
| VAF | forward early voltage (= 50.0) |
| VAR | reverse early voltage (= 1.0e20) |
| Vt | kt/q (= 0.002585 at 27 deg. C) |

The DC nodal voltages of the common-emitter circuit were determined using a FORTRAN sub-routine, then SPICE for verification. The results are as follows:

| node | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|
| custom | 0.9972 | 5.0000 | 4.086 | 0.2311 | 0.9963 | 4.086 |
| SPICE | 0.9973 | 5.0000 | 4.087 | 0.2308 | NA | 4.087 |

Minor discrepancies are likely due to the assumed values for zero and infinite resistance. Otherwise, the values are in perfect agreement and indicate that the equation solving works. The SPICE runs were performed using *IG-SPICE* on a SUN-4 and *P-SPICE* on a PC. The SPICE results exactly agreed with each other.

### 2.1B. Simulation of Catastrophic Faults

Catastrophic (i.e., hard) faults have been modeled through Monte Carlo simulation for the one-stage amplifier circuit described above. The analysis neglected multiple faults; that is, only a single catastrophic fault was introduced per simulation run. The induced faults were catastrophic resistor failures that represent transistor failure. In particular the resistances $R_b$, $R_c$, and $R_e$, which are nominally 0.1 Ω, were assigned relatively large values of 1 MegΩ, while the resistances $R_{bc}$, $R_{be}$, and $R_{ce}$, which are nominally infinite, were assigned relatively small values of 10 MΩ. Table 2-1A lists the summary Monte Carlo statistics for each of the six DC voltages, and it can be used to compare faulted with nonfaulted behavior per measured DC voltage.

Cursory examination of Table 2-1A reveals quite noticeable differences among the various groups and suggests that statistical discrimination techniques would be effective not only in discriminating between faulted and non-faulted behavior, but also in discriminating among the faulted groups. Such an analysis was performed using a the DISCRIM procedure in SAS. The results of this analysis, which we will now describe, show perfect discrimination for submitted test sets.

### 2.1C. Fault Discrimination

SAS's DISCRIM routine was used to discriminate among the catastrophic faults and to assign them to specific fault groups. Given a training set of data that is multivariate normal, the DISCRIM procedure computes a discrimination criterion based on a multivariate measure of the generalized squared distance (see Part 1, eq. 1-8).

Our training set consisted of 2000 simulations per fault group. Each simulation run resulted in a multivariate observation vector consisting of three DC voltages $V_3$, $V_4$, and $V_5$ (we note that correlations among the DC voltages reduced the relevant number of voltages from 6 to 3). The SAS

# Table 2-1A. Summary Statistics for One-Stage Amplifier with Catastrophic Faults

| VARIABLE | N | MEAN | STDEV | MINIMUM | MAXIMUM | RANGE | SKEWNESS | KURTOSIS |
|---|---|---|---|---|---|---|---|---|
| V1 | | | | | | | | |
| non-faulted | 3000 | 1.00 | 0.08 | 0.74 | 1.33 | 0.59 | 0.36 | 0.62 |
| RB1 | 3000 | 1.12 | 0.12 | 0.75 | 1.61 | 0.86 | 0.29 | 0.30 |
| RC1 | 3000 | 0.71 | 0.01 | 0.67 | 0.74 | 0.07 | -0.21 | 0.36 |
| RE1 | 3000 | 1.12 | 0.12 | 0.75 | 1.62 | 0.87 | 0.29 | 0.29 |
| RBC1 | 3000 | 1.32 | 0.36 | 0.90 | 1.92 | 1.02 | -0.24 | -1.80 |
| RBE1 | 3000 | 0.03 | 0.00 | 0.02 | 0.04 | 0.03 | 0.45 | 0.35 |
| RCE1 | 3000 | 1.12 | 0.12 | 0.75 | 1.62 | 0.87 | 0.28 | 0.25 |
| V2 | | | | | | | | |
| non-faulted | 3000 | 5.00 | 0.00 | 5.00 | 5.00 | 0.00 | N/A | N/A |
| RB2 | 3000 | 5.00 | 0.00 | 5.00 | 5.00 | 0.00 | N/A | N/A |
| RC2 | 3000 | 5.00 | 0.00 | 5.00 | 5.00 | 0.00 | N/A | N/A |
| RE2 | 3000 | 5.00 | 0.00 | 5.00 | 5.00 | 0.00 | N/A | N/A |
| RBC2 | 3000 | 5.00 | 0.00 | 5.00 | 0.00 | 0.00 | N/A | N/A |
| RBE2 | 3000 | 5.00 | 0.00 | 5.00 | 0.00 | 0.00 | N/A | N/A |
| RCE2 | 3000 | 5.00 | 0.00 | 5.00 | 5.00 | 0.00 | -31.59 | 996.66 |
| V3 | | | | | | | | |
| non-faulted | 3000 | 4.08 | 0.31 | 2.36 | 4.90 | 2.54 | -0.65 | 1.03 |
| RB3 | 3000 | 4.96 | 0.01 | 4.87 | 4.99 | 0.12 | -0.68 | 0.92 |
| RC3 | 3000 | 4.99 | 0.00 | 4.99 | 5.00 | 0.01 | 0.67 | 5.18 |
| RE3 | 3000 | 5.00 | 0.00 | 5.00 | 5.00 | 0.00 | -1.59 | 0.54 |
| RBC3 | 3000 | 2.81 | 1.42 | 1.35 | 4.50 | 3.15 | 0.34 | -1.88 |
| RBE3 | 3000 | 5.00 | 0.00 | 5.00 | 5.00 | 0.00 | 15.72 | 245.41 |
| RCE3 | 3000 | 1.03 | 0.12 | 0.69 | 1.52 | 0.83 | 0.22 | 0.10 |

KEY

| | NOMINAL VALUES | FAULTED VALUES |
|---|---|---|
| RB | 0.1 Ohm | 1 MOhm |
| RC | 0.1 Ohm | 1 MOhm |
| RE | 0.1 Ohm | 1 MOhm |
| RBC | infinite | 10 Ohm |
| RBE | infinite | 10 Ohm |
| RCE | infinite | 10 Ohm |

# Table 2-1A. (cont.) Summary Statistics for One-Stage Amplifier with Catastrophic Faults

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| V4 | | | | | | | | |
| non-faulted | 3000 | 0.23 | 0.07 | 0.03 | 0.54 | 0.52 | 0.49 | 0.69 |
| RB4 | 3000 | 0.01 | 0.00 | 0.00 | 0.03 | 0.03 | 0.67 | 0.76 |
| RC4 | 3000 | 0.01 | 0.00 | 0.00 | 0.02 | 0.02 | 0.20 | 0.84 |
| RE4 | 3000 | 0.55 | 0.12 | 0.20 | 1.03 | 0.82 | 0.33 | 0.31 |
| RBC4 | 3000 | 0.53 | 0.35 | 0.12 | 1.12 | 1.00 | -0.23 | -1.80 |
| RBE4 | 3000 | 0.02 | 0.00 | 0.01 | 0.04 | 0.03 | 0.44 | 0.34 |
| RCE4 | 3000 | 1.00 | 0.11 | 0.66 | 1.48 | 0.82 | 0.21 | 0.10 |
| V5 | | | | | | | | |
| non-faulted | 3000 | 1.00 | 0.08 | 0.74 | 1.33 | 0.59 | 0.36 | 0.62 |
| RB5 | 3000 | 0.70 | 0.01 | 0.65 | 0.74 | 0.09 | -0.08 | 0.25 |
| RC5 | 3000 | 0.71 | 0.01 | 0.67 | 0.74 | 0.07 | -0.21 | 0.36 |
| RE5 | 3000 | 1.12 | 0.12 | 0.75 | 1.62 | 0.87 | 0.29 | 0.29 |
| RBC5 | 3000 | 1.30 | 0.38 | 0.86 | 1.92 | 1.06 | -0.25 | -1.81 |
| RBE5 | 3000 | 0.03 | 0.00 | 0.02 | 0.04 | 0.03 | 0.45 | 0.35 |
| RCE5 | 3000 | 1.12 | 0.12 | 0.75 | 1.62 | 0.87 | 0.28 | 0.25 |
| V6 | | | | | | | | |
| non-faulted | 3000 | 4.08 | 0.31 | 2.36 | 4.90 | 2.54 | -0.65 | 1.03 |
| RB6 | 3000 | 4.96 | 0.01 | 4.87 | 4.99 | 0.12 | -0.68 | 0.92 |
| RC6 | 3000 | 0.03 | 0.00 | 0.02 | 0.10 | 0.08 | 10.52 | 146.44 |
| RE6 | 3000 | 5.00 | 0.00 | 5.00 | 5.00 | 0.00 | -1.59 | 0.54 |
| RBC6 | 3000 | 2.81 | 1.42 | 1.35 | 4.50 | 3.15 | 0.34 | -1.88 |
| RBE6 | 3000 | 5.00 | 0.00 | 5.00 | 5.00 | 0.00 | 15.72 | 245.41 |
| RCE6 | 3000 | 1.03 | 0.12 | 0.69 | 1.52 | 0.83 | 0.22 | 0.10 |

**KEY**

| | NOMINAL VALUES | FAULTED VALUES |
|---|---|---|
| RB | 0.1 Ohm | 1 MOhm |
| RC | 0.1 Ohm | 1 MOhm |
| RE | 0.1 Ohm | 1 MOhm |
| RBC | infinite | 10 Ohm |
| RBE | infinite | 10 Ohm |
| RCE | infinite | 10 Ohm |

test of homogeneity of individual within-group covariances revealed a lack of homogeneity; hence, the within-group covariance matrices were used to construct the discriminant function, rather than the pooled covariance matrix. This resulted in an asymmetric distance function, as shown in the SAS summary table (Table 2-1B).

Out test set consisted of 1000 simulations per group, generated using a different random number seed to ensure that the test and training sets were different. Table 2-1B, showing SAS output, indicates that none of the test events were misclassified.

### 2.1D. Discrimination and Classification of Parametric Faults

Parametric (soft) fault analyses have been performed for the single one-stage amplifier described above. The present discussion applies to DC voltages only. The primary goal of these analyses was to determine the "fault continua" of circuit parameters and to determine thresholds along these continua where soft faults begin to have a detectable difference upon the nominal DC circuit performance; that is, to perform simple discrimination between nominal and soft-faulted circuits. (Nominal performance here refers to performance of the circuit given stochastic variations of the circuit parameters over their tolerance ranges, and is characterized by a cluster of points in a multivariate voltage response space.) Thus, given parametric deviations of a circuit parameter from its nominal value (and outside its specified tolerance range), we also attempted to partition each of the fault continua into discrete classes for actual classification of soft-fault behaviors of different circuit parameters, as we were so successful in doing previously for catastrophic faults.

We initially chose to fault the transistor base-emitter leakage resistor $R_{be}$ parametrically. Under nominal conditions, this leakage resistance is for practical purposes infinite (actually, $1 \times 10^{10}\ \Omega$ in our simulations), while under catastrophic fault conditions it is near zero (actually $10.0\ \Omega$ in our simulations). Soft-fault data were generated by varying the resistance in powers of 10 from 100.0 $\Omega$ to $1 \times 10^{9}\ \Omega$. Five hundred (500) values were generated for each of 8 soft-fault classes: $R_{be2}$ (100 $\Omega$) to $R_{be9}$ ($1 \times 10^{9}\ \Omega$), while "Good" (nominal) and "Hard" fault classes had 512 values each (these two classes contained more values because they came from prior analyses; they were generated, however, with the same random number seed).

**Table 2-1B. Explanation of SAS Discrimination Analysis Table**
(table given on following page)

Table 2-1B gives the output from a typical SAS discrimination analysis. The input file for the analysis is shown below the table.

Looking across the table, we have the classification categories of the faults, namely "good", "faulted $R_b$," "faulted $R_{bc}$," etc. Looking down the table, we have the type of fault applied. For each fault type applied, the first row gives the number of simulations classified under each fault category; the second row gives the percentage of the number of simulations classified under each fault category. To illustrate the use of the table, consider the faulting of $R_{bc}$, where this component takes on the value of 10 ohms when faulted (from a nominal infinite ohms). 1000 Monte Carlo simulations were applied to the circuit under this fault condition, and all 1000 results were classified under the category of "$R_{bc}$ fault." In other words, 100% of the simulations where $R_{bc}$ is faulted and all remaining components have nominal variations were properly classified. Likewise, all other fault conditions (including "good" operation) were 100% properly classified.

This result shows that catastrophic faults in the one-stage amplifier can easily be classified, assuming the restrictions to the designated fault types, the accessibility to the voltages of the analysis, and the nominal component variation used in the simulations. However, small shifts in component parameters, i.e., parametric drift, will likely cause discrimination error.

# Table 2-1B. Catastrophic Faults

SAS 10:03 THURSDAY, DECEMBER 21, 1989 14

DISCRIMINANT ANALYSIS CLASSIFICATION SUMMARY FOR TEST DATA: WORK.TEST

GENERALIZED SQUARED DISTANCE FUNCTION

$$D_J^2(X) = (X-\bar{X}_J)' \, COV_J^{-1} \, (X-\bar{X}_J) + LN\,|COV_J|$$

POSTERIOR PROBABILITY OF MEMBERSHIP IN EACH FAULT.

$$PR(J|X) = EXP(-.5\, D_J^2(X)) \,/\, SUM_K\, EXP(-.5\, D_K^2(X))$$

NUMBER OF OBSERVATIONS AND PERCENTS CLASSIFIED INTO FAULT

| FROM FAULT | good | rb | rbc | rbe | rc | rce | re | TOTAL |
|---|---|---|---|---|---|---|---|---|
| good | 1000 | 0 | 0 | 0 | 0 | 0 | 0 | 1000 |
| | 100.00 | 0.00 | 0 00 | 0.00 | 0 00 | 0.00 | 0 00 | 100.00 |
| rb | 0 | 1000 | 0 | 0 | 0 | 0 | 0 | 1000 |
| | 0.00 | 100.00 | 0 00 | 0.00 | 0.00 | 0.00 | 0.00 | 100.00 |
| rbc | 0 | 0 | 1000 | 0 | 0 | 0 | 0 | 1000 |
| | 0 00 | 0 00 | 100 00 | 0 00 | 0 00 | 0.00 | 0 00 | 100 00 |
| rbe | 0 | 0 | 0 | 1000 | 0 | 0 | 0 | 1000 |
| | 0.00 | 0 00 | 0 00 | 100.00 | 0.00 | 0.00 | 0 00 | 100 00 |
| rc | 0 | 0 | 0 | 0 | 1000 | 0 | 0 | 1000 |
| | 0 00 | 0 00 | 0 00 | 0.00 | 100 00 | 0.00 | 0 00 | 100.00 |
| rce | 0 | 0 | 0 | 0 | 0 | 1000 | 0 | 1000 |
| | 0.00 | 0.00 | 0 00 | 0 00 | 0.00 | 100.00 | 0 00 | 100.00 |
| re | 0 | 0 | 0 | 0 | 0 | 0 | 1000 | 1000 |
| | 0 00 | 0 00 | 0 00 | 0 00 | 0 00 | 0.00 | 100 00 | 100.00 |
| TOTAL | 1000 | 1000 | 1000 | 1000 | 1000 | 1000 | 1000 | 7000 |
| PERCENT | 14 29 | 14.29 | 14.29 | 14.29 | 14.29 | 14 29 | 14.29 | 100 00 |
| PRIORS | 0 1429 | 0.1429 | 0 1429 | 0.1429 | 0 1429 | 0.1429 | 0 1429 | |

```
data Faults,
 infile '(alexg.circuit.discrim)fault dat',
 input Fault $ v1 v2 v3 v4 v5 v6,

data Test,
 infile '(alexg.circuit.discrim)test dat',
 input Fault $ v1 v2 v3 v4 v5 v6,

Proc Discrim Data=Faults S Pool=Test Wcov Wcorr Pcov Pcorr Out=FaultCal,
  Class Fault
  Var v4 v5 v6,

Proc Discrim Data=FaultCal Testdata=Test TestListErr,
  Class Fault,
  Testclass Fault,
  Var v4 v5 v6,
```

These data were used as a training set for DISCRIM. After applying the training set data to test the discrimination, we arrived at the results shown in Table 2-1C. For the "harder" soft-fault classes (i.e., those classes with resistance values less than or equal to $1x10^4$ Ω; namely, the four fault classes Hard, $R_{be2}$, $R_{be3}$, and $R_{be4}$), the classifier succeeds, for the most part, in unambiguously classifying the faults into their appropriate fault classes. For the "softer" faults (i.e., those above $1x10^4$ ohms; namely, the six classes Good and $R_{be5}$ to $R_{be9}$), however, the classifier function performs poorly and seems to want to classify the majority of the faulted circuits within the $1x10^6$ Ω class. The reason for the poor performance becomes apparent when one examines the multivariate means and variances of the soft-fault clusters. For these softer-fault classes, there is very little numerical difference in the means and variances. The classifier appears to be choosing most often that cluster which best represents the 6 clusters.

A soft-fault analysis was also performed for the transistor's forward beta (BF) as one representation of transistor gain change. Here five fault classes were chosen: Class 1 for catastrophic low (BF values from 0 to 40); Class 2 for soft low (BF values from 40 to 70); Class 3 for nominal (BF from 70 to 90); Class 4 for soft high (BF from 90 to 120); and Class 5 for catastrophic high (BF from 120 to 200). Five hundred voltage sets were generated in each class. The SAS discrimination results are shown in Table 2-1D. Once again, the classifier function works better on the high and low catastrophic faults than it does on the softer faults, although the overall performance of the classifier is poor.

To see how this classifier would work when an independent set of data is used as the test data, we generated test data consisting of 20 points for each integer value of BF between 1 and 200, for a total of 400 test data points. The SAS results for this test are shown in Table 2-1E. The classification results are what would be expected from the poor performance of the calibration data; hence, no improvement in discrimination was obtained.

# Table 2-1C. Parametric Faulting of Resistor $R_{be}$

SAS DISCRIM CLASSIFICATION SUMMARY FOR RBE CALIBRATION DATA: WORK.FAULTS

GENERALIZED SQUARED DISTANCE FUNCTION:

$$D^2_J(X) = (X-\bar{X}_J)' \, COV_J^{-1} \, (X-\bar{X}_J) + LN\,|COV_J|$$

POSTERIOR PROBABILITY OF MEMBERSHIP IN EACH FAULT:

$$PR(J|X) = EXP(-.5\,D^2_J(X)) \,/\, SUM_K\, EXP(-.5\,D^2_K(X))$$

NUMBER OF OBSERVATIONS AND PERCENTS CLASSIFIED INTO FAULT.

| FROM FAULT | good | hard | rbe2 | rbe3 | rbe4 | rbe5 | rbe6 | rbe7 | rbe8 | rbe9 | TOTAL |
|---|---|---|---|---|---|---|---|---|---|---|---|
| good | 5 | 0 | 0 | 0 | 0 | 26 | 275 | 203 | 3 | 0 | 512 |
| | 0.98 | 0.00 | 0.00 | 0.00 | 0.00 | 5.08 | 53 71 | 39.65 | 0.59 | 0.00 | 100.00 |
| hard | 0 | 510 | 0 | 0 | 0 | 2 | 0 | 0 | 0 | 0 | 512 |
| | 0.00 | 99.61 | 0.00 | 0.00 | 0.00 | 0.39 | 0.00 | 0 00 | 0.00 | 0.00 | 100 00 |
| rbe2 | 0 | 0 | 498 | 0 | 0 | 2 | 0 | 0 | 0 | 0 | 500 |
| | 0.00 | 0.00 | 99.60 | 0.00 | 0.00 | 0.40 | 0.00 | 0.00 | 0.00 | 0.00 | 100.00 |
| rbe3 | 0 | 0 | 0 | 498 | 0 | 2 | 0 | 0 | 0 | 0 | 500 |
| | 0.00 | 0.00 | 0.00 | 99.60 | 0 00 | 0.40 | 0.00 | 0.00 | 0.00 | 0.00 | 100.00 |
| rbe4 | 0 | 0 | 0 | 0 | 495 | 5 | 0 | 0 | 0 | 0 | 500 |
| | 0.00 | 0.00 | 0 00 | 0.00 | 99.00 | 1.00 | 0.00 | 0 00 | 0 00 | 0.00 | 100 00 |
| rbe5 | 0 | 0 | 0 | 0 | 0 | 83 | 340 | 75 | 0 | 2 | 500 |
| | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 16.60 | 68 00 | 15 00 | 0.00 | 0.40 | 100.00 |
| rbe6 | 3 | 0 | 0 | 0 | 0 | 30 | 277 | 186 | 1 | 3 | 500 |
| | 0.60 | 0.00 | 0.00 | 0.00 | 0.00 | 6.00 | 55.40 | 37.20 | 0.20 | 0.60 | 100.00 |
| rbe7 | 4 | 0 | 0 | 0 | 0 | 28 | 268 | 197 | 3 | 0 | 500 |
| | 0.80 | 0.00 | 0.00 | 0.00 | 0.00 | 5.60 | 53.60 | 39.40 | 0.60 | 0.00 | 100.00 |
| rbe8 | 4 | 0 | 0 | 0 | 0 | 28 | 266 | 199 | 3 | 0 | 500 |
| | 0.80 | 0.00 | 0.00 | 0.00 | 0.00 | 5.60 | 53.20 | 39 80 | 0.60 | 0.00 | 100.00 |
| rbe9 | 4 | 0 | 0 | 0 | 0 | 25 | 267 | 200 | 3 | 1 | 500 |
| | 0.80 | 0.00 | 0 00 | 0.00 | 0 00 | 5.00 | 53 40 | 40 00 | 0 60 | 0 20 | 100 00 |
| TOTAL | 20 | 510 | 498 | 498 | 495 | 231 | 1693 | 1060 | 13 | 6 | 5024 |
| PERCENT | 0 40 | 10.15 | 9 91 | 9.91 | 9.85 | 4 60 | 33 70 | 21 10 | 0.26 | 0 12 | 100 00 |
| PRIORS | 0.1000 | 0 1000 | 0.1000 | 0.1000 | 0 1000 | 0 1000 | 0 1000 | 0 1000 | 0 1000 | 0 1000 | |

# Table 2-1D. Parametric Faulting of Forward Beta Parameter BF
(testing performed using training set data)

SAS DISCRIM CLASSIFICATION SUMMARY FOR BF CALIBRATION DATA: WORK FAULTS

GENERALIZED SQUARED DISTANCE FUNCTION

$$D_J^2(X) = (X-\bar{X}_J)' \, COV_J^{-1} \, (X-\bar{X}_J) + LN\,|COV_J|$$

POSTERIOR PROBABILITY OF MEMBERSHIP IN EACH FAULT

$$PR(J|X) = EXP(-.5\,D_J^2(X)) \,/\, SUM_K\, EXP(-.5\,D_K^2(X))$$

NUMBER OF OBSERVATIONS AND PERCENTS CLASSIFIED INTO FAULT

| FROM FAULT | 1 | 2 | 3 | 4 | 5 | TOTAL |
|---|---|---|---|---|---|---|
| 1 | 361 | 130 | 2 | 2 | 5 | 500 |
| | 72.20 | 26 00 | 0.40 | 0.40 | 1.00 | 100.00 |
| 2 | 85 | 292 | 17 | 47 | 59 | 500 |
| | 17 00 | 58.40 | 3 40 | 9 40 | 11 80 | 100 00 |
| 3 | 66 | [illegible]44 | 34 | 50 | 106 | 500 |
| | 13.20 | 48 80 | 6.80 | 10.00 | 21.20 | 100.00 |
| 4 | 46 | 191 | 30 | 90 | 143 | 500 |
| | 9.20 | 38 20 | 6.00 | 18.00 | 28 60 | 100 00 |
| 5 | 32 | 166 | 27 | 88 | 187 | 500 |
| | 6.40 | 33 20 | 5.40 | 17.60 | 37 40 | 100.00 |
| TOTAL | 590 | 1023 | 110 | 277 | 500 | 2500 |
| PERCENT | 23.60 | 40.92 | 4.40 | 11.08 | 20 00 | 100 00 |
| PRIORS | 0 2000 | 0.2000 | 0.2000 | 0.2000 | 0 2000 | |

# Table 2-1E. Parametric Faulting of Forward Beta Parameter BF
(testing performed with independent test data)

SAS DISCRIM CLASSIFICATION SUMMARY FOR BF TEST DATA: WORK TEST

GENERALIZED SQUARED DISTANCE FUNCTION:

$$D^2_J(X) = (X-\bar{X}_J)' \, COV_J^{-1} \, (X-\bar{X}_J) + LN\,|COV_J|$$

POSTERIOR PROBABILITY OF MEMBERSHIP IN EACH FAULT:

$$PR(J|X) = EXP(-.5\,D^2_J(X)) \,/\, SUM_K\, EXP(-.5\,D^2_K(X))$$

NUMBER OF OBSERVATIONS AND PERCENTS CLASSIFIED INTO FAULT:

| FROM FAULT | 1 | 2 | 3 | 4 | 5 | TOTAL |
|---|---|---|---|---|---|---|
| 1 | 532 | 235 | 6 | 4 | 3 | 780 |
| | 68.21 | 30 13 | 0 77 | 0 51 | 0.38 | 100.00 |
| 2 | 97 | 354 | 36 | 41 | 72 | 600 |
| | 16.17 | 59.00 | 6.00 | 6.83 | 12.00 | 100.00 |
| 3 | 45 | 196 | 25 | 55 | 79 | 400 |
| | 11.25 | 49.00 | 6.25 | 13.75 | 19.75 | 100.00 |
| 4 | 65 | 249 | 44 | 87 | 155 | 600 |
| | 10.83 | 41.50 | 7.33 | 14.50 | 25.83 | 100.00 |
| 5 | 112 | 521 | 98 | 264 | 625 | 1620 |
| | 6 91 | 32.16 | 6.05 | 16 30 | 38 58 | 100.00 |
| TOTAL | 851 | 1555 | 209 | 451 | 934 | 4000 |
| PERCENT | 21.28 | 38.88 | 5.23 | 11.28 | 23.35 | 100.00 |
| PRIORS | 0.2000 | 0.2000 | 0.2000 | 0.2000 | 0 2000 | |

### 2.1E. Effect of Discriminator Training Set Size

To improve the efficiency of the training set construction, we have investigated the effect of reducing the training set size.

Table 2-1F shows the effects of reducing training set size. This table was constructed by applying one of five possible catastrophic faults to the one-stage amplifier, as described in Section 2.1B. These faults were: base open, collector open, emitter open, base-collector short, base-emitter short emitter-collector short. The number of incorrect classifications are given in Table 2-1F. "Training set size" refers to the number of sets of data per fault type (from Monte Carlo runs) that were used to generate the discrimination function. "Validation size" is the number of sets of data, for each fault type and from Monte Carlo runs, applied to test the discriminator.

**Table 2-1F. Effect of Training Set Size**

| training set size / validation size | 10 | 50 | 200 |
|---|---|---|---|
| 10 | 0* | 0 | 0 |
| 50 | 0 | 0* | 0 |
| 200 | 2 | 0 | 1* |

(* test and validation data sets were the same,otherwise they were different data sets)

We can see from Table 2-1F that a reasonable amount of fault classification accuracy has occurred even for only 10 training sets (~ 99% accuracy). We will therefore have considered using reduced training set sizes for the larger circuits, where more data must be analyzed.

### 2.1F. Comparison of DC and AC Results

During realistic IC probing measurements, the intrinsic nodes of a transistor may often not be available for measurement. In fact, the notion of series lead resistance is only conceptual in many IC transistors. A chip's series resistance may actually be for modeling convenience only, and reflect resistances internal to the transistor (e.g., channel and metallization resistances). In other cases, the "lead" resistances are true series resistances that cannot be fully probed. For the latter case, we have considered the situation where only a limited number of voltage nodes are available for probing.

Referring to the one-stage amplifier of Figure 2-1A, we now assume that only the points $V_{in}$,-$V_2$,$V_3$, $V_4$, and $V_{supply}$ are available for both AC and DC probing. Figure 2-1B shows the AC small-signal model used for the simulations, which follows that of SPICE. Following the SAS discrimination procedures discussed in our earlier reports, the results are now more disappointing, yet instructive.

Faults were simulated by applying the component values shown in Table 2-1G, one at a time, to the circuit simulations. The simulations made use of our custom simulation program described in earlier reports. A variety of results occurred, including numerical instabilities in the DC solution for the cases of $R_2$ shorted, $R_3$ opened, and $R_e$ opened. We therefore dropped these cases from the analysis (which is not a serious limitation, since in reality the circuit would obviously not work). The AC solutions failed for the cases $R_1$ open and $R_{bc}$ shorted -- again an obvious failure during measurement. Also note that $R_c$ had to be set to $10^3$ to emulate a faulted condition; "opening" this value to $10^{10}$ caused numerical problems as well. 100 runs for each fault type were typically applied. This value was used in view of our earlier results.

***Results:***

*DC Only Analysis:*

The operating conditions were $V_{supply}$ = 5V, $R_1$ = 1 megohm. Nominal variations of all components were 10% (for 1 sigma). Applying only the DC points of $V_{in}$, $V_2$, and $V_3$ to SAS resulted in the failure to classify 100% of the "Good" cases correctly. In other words, runs of Good circuits (no faults present) could not be distinguished from certain faulted cases.

*AC analysis:*

This analysis applied a 100 Hz, 1V signal to the circuit, and included, in addition to the DC nodes above, the real and imaginary parts of the AC nodal voltages from nodes $V_{in}$, $V_2$, $V_3$, and $V_4$. SAS analysis resulted in correct classification of 12% of the Good cases. Thus the introduction of AC measurements somewhat improved the classification, although it was still poor.

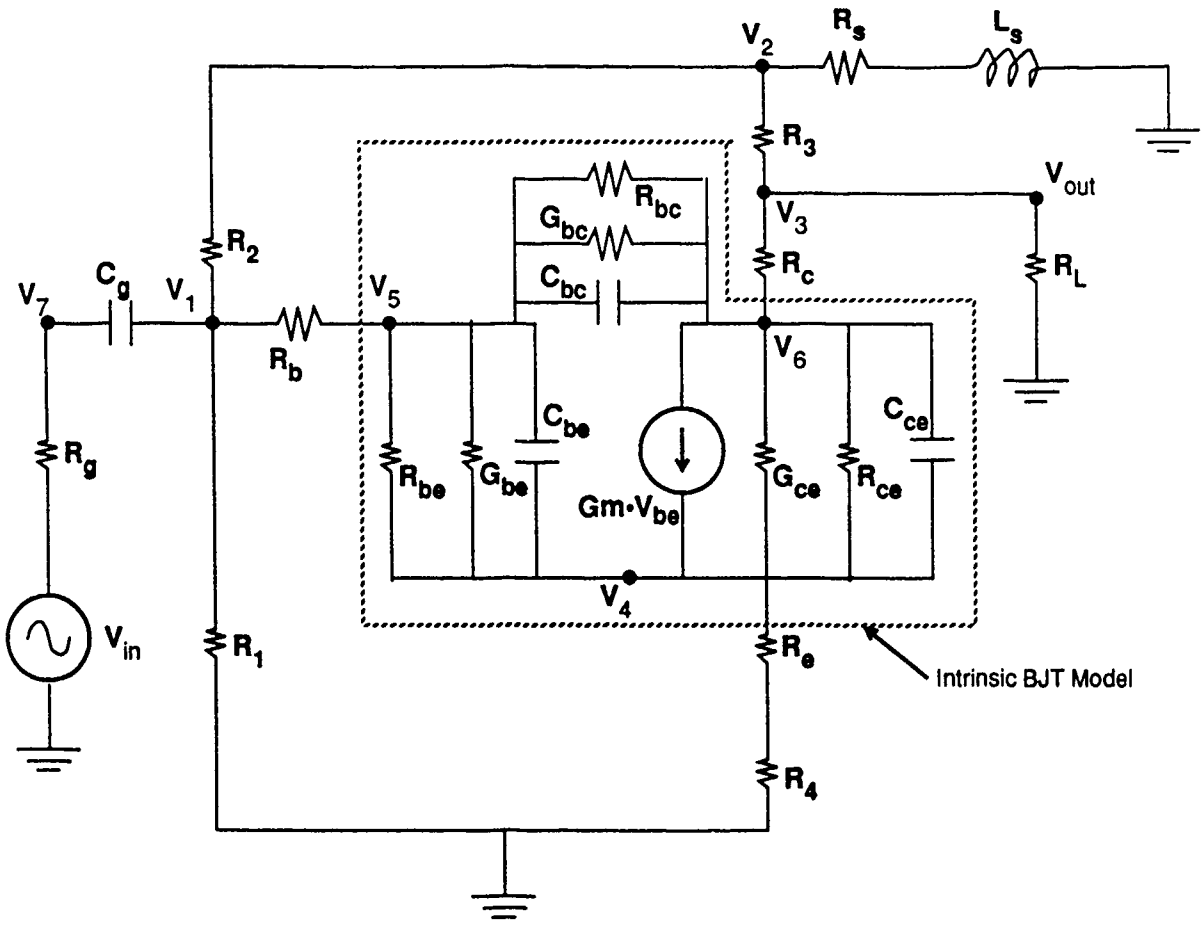



***Figure 2-1B.*** AC analysis of one-stage amplifier shown in Figure 2-1A.

**Table 2-1G. Catastrophic Fault Simulations**
(refer to components in Figure 2-1A;
all values in ohms)

| Component | Fault | Value |
|---|---|---|
| $R_1$ | short | $10^{-3}$ |
| $R_1$ | open | $10^{10}$ |
| $R_2$ | short | $10^{-3}$ |
| $R_2$ | open | $10^{10}$ |
| $R_3$ | short | $10^{-3}$ |
| $R_3$ | open | $10^{10}$ |
| $R_3$ | short | $10^{-3}$ |
| $R_3$ | open | $10^{10}$ |
| $R_b$ | open | $10^{10}$ |
| $R_c$ | open | $10^{10}$ |
| $R_e$ | open | $10^{10}$ |
| $R_{be}$ | short | $10^{-3}$ |
| $R_{bc}$ | short | $10^{-3}$ |
| $R_{ce}$ | short | $10^{-3}$ |
| BF | low | 10 |
| BF | high | 50 |
| VAF | low | 5 |
| VAF | high | 95 |
| FC | low | 0.05 |
| FC | high | .95 |

In both the AC and DC cases, most of the faults were incorrectly classified into the categories of FC (the forward bias depletion capacitor coefficient) low and high. In other words, altering this parameter had minimal effect on the performance of Good circuits. This is to be expected, since at 100 Hz, capacitive effects are effectively undetectable. At higher frequencies, this is would not be the case. In fact, reducing the component tolerances to effectively no randomization (zero tolerance) yielded 100% classification of the good cases as a "FC low", further confirming our notion that FC has no effect on the circuit operation.

Since faulting FC had no effect on the circuit operation, we removed it from subsequent analyses. For zero component tolerances, 100% of the Good circuits were then correctly classified. However, as the component tolerances were increased to 10%, only 44% of the Good circuits were correctly classified as "Good." 40% of the Good circuits were misclassified as "open $R_c$," 8% as "BF low," and 8% as "VAF high." We therefore conclude that, *given the operating conditions of the circuit and the test nodes*, we cannot effectively separate the faulted cases from the non-faulted cases. Obviously a better choice of operating conditions is needed for better fault discrimination (more on this below).

For curiosity, we eliminated all faults that did not correspond to resistors opening or shorting, and used a component tolerance of 10%. Discriminant analysis resulted in 60% of the Good circuits being classified as "Good," with the remaining 40% incorrectly classified as "$R_c$ open." Once again we have a situation where now a DC component change has minimal effect on circuit operation, which leads to difficult fault discrimination. Section 2.1G of this report will discuss this matter in more detail.

Finally, in order to check if SAS uses order as a classification criteria when two classes of data are identical, we relabeled class "Good" as "0Good" to force the label of the "Good" class to alphabetically precede all other fault classes. With a 10% tolerance on all components, there was no difference in the classification regardless of the placement of the Good category in the labeling (i.e., only 12% of the Good circuits were classified as "Good"). However, when the tolerance was reduced to zero, the original and erroneous classification of all Good circuits as "FC low" disappeared, and all Good circuits became classified as "Good"! In other words, when SAS cannot distinguish between identical classes of data, it simply chooses the class corresponding to the alpha-

betically first label. In any event, the importance of this exercise is to realize that when certain types of simulated faults produce circuit responses that cannot be distinguished from normal (Good) circuits, classification may become extremely erroneous.

### 2.1G. Response Screening

In larger circuits where there are many nodes operating over many conditions, much of the data to be used in a discrimination analysis may exhibit linear dependencies that cause the discrimination computations to be much longer than they have to. In other words, much of the data may be redundant and therefore can be eliminated.

To investigate this concept, the SAS input data from the Monte Carlo analyses was ran through the computation of partial correlation coefficients. Variable pairs were eliminated that had high correlations. In the present example, and choosing correlation coefficients of 0.8 or greater as the criteria for elimination, we eliminated the imaginary parts of $V_{in}$, $V_2$, and $V_3$. Running an analysis where FC was not faulted yielded a degradation of our Good classification of 44% to 20%. We conclude that the present model may need all the nodes, and more, for proper fault discrimination. For other circuits, however, this technique still might be useful.

Note that the results presented herein are poor compared to the typically 98 to 100% proper classification of faulted and Good circuits described in our earlier reports. In the earlier work, all circuit nodes were probed. This illustrates the importance of obtaining as much data as possible for accurate fault detection.

### 2.1H. Effect of Collector Resistance Variations on Fault Detection.

During our discriminant analysis, we observed a consistent misclassification of Good circuits as falling into the "$R_c$ open" class. (Also recall that we could not fully open $R_c$ to $10^{10}$ ohms, and had to restrict the open circuit emulation to $10^3$ ohms.) It turns out that if the device is operating in saturation for a given base-emitter voltage, the current flow through $R_3$ will remain constant since the transistor acts as a constant current source. In other words, the voltage drop from nodes 2 to 3 (see Figure 2-1C) will remain constant over a large range of $R_c$. Since we are using either

$V_2$ or $V_3$ (and not $V_c$), detection of changes in $R_c$ is difficult at best. One possible method of detection is to ensure that a full open at $R_c$ draws the drain current to zero, thereby changing the voltage drop $V_2$ - $V_3$ (for a very large $R_L$). In other words, the load line for the collector would intersect the I-V characteristics of the device at collector voltages below the "knee." Unfortunately, instabilities in the numerical methods precluded this effect. However, this problem was overcome by using SPICE, as will be discussed in the next section.

### 2-1I. Single-Stage Amplifier Revisited (SFA Program Used)

To verify the operation of the Statistical Fault Analyzer (SFA), we once again looked at the one-stage amplifier of our earlier work. Figure 2-1D shows the circuit simulation input file, and Table 2-1H summarizes the applied faults. Note that in this run, faults were at the transistor only. Table 2-1I summarizes the SAS DISCRIM results when looking only at the DC voltages of nodes Vin, $V_2$, $V_3$, and $V_4$ (see Figure 2-1A). Table 2-1I indicates severe misclassification among the "nominal," "Q_HI" (beta high), and "RB short" categories; i.e., the Q_HI and RB short categories cannot be readily distinguished from the nominal runs.

Table 2-1J shows the result of the SAS DISCRIM analysis where the AC voltages at nodes labeled $V_2$, $V_3$, and $V_4$ (Figure 2-1A) are included along with the DC values. We now observe almost perfect classification of all applied faults. In other words, the methods that we have used to fault $R_b$ and B[illegible] a affect the AC characteristics of the device much more than the DC characteristics. This points to the necessity of applying appropriate test signals to aid in the discrimination.

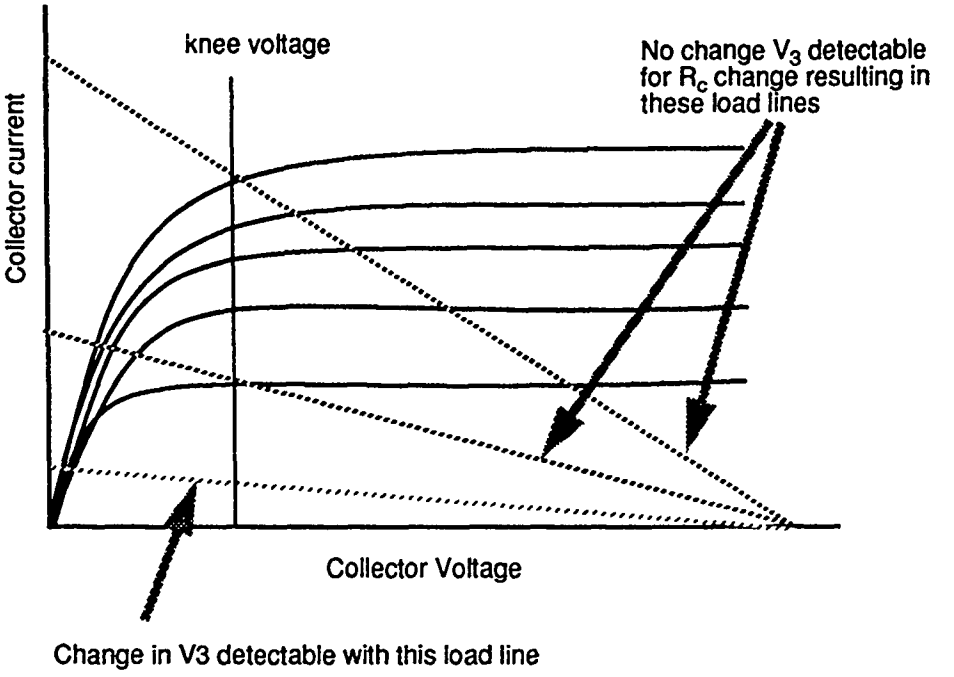



***Figure 2-1C.*** Effect of changing load lines on the ability to detect changes in $V_3$.

**Table2-1H. Catastrophic Fault Simulations**
(refer to components in Figure 2-1D;
all resistance values in ohms)

| Component | Fault | Value |
|---|---|---|
| RC | open | $10^6$ |
| RE | open | $10^6$ |
| RB | open | $10^6$ |
| RB | short | 1.0 |
| RBC | short | 1.0 |
| RBE | short | 1.0 |
| RCE | short | 1.0 |
| BF (QNL) | low | 1.0 |
| BF (QNL) | high | 200.0 |

```
Single-stage amplifier
#GOOD 200
#RC 1.0e6 200 RC_OPEN
#RE 1.0e6 200 RE_OPEN
#RB 1.0e6 200 RB_OPEN
#RB 1.0 200 RB_Short
#Rbc 1.0 200 RBC_Short
#Rbe 1.0 200 RBE_Short
#Rce 1.0 200 RCE_Short
#QNL 1.0 200 Q_low BF
#QNL 200.0 200 Q_hi BF
%dc 1 2 3 7
%ac 2 3 7
VCC 8 0 5.0
RSDC 8 2 1.0
VIN 10 0 AC 1
CIN 10 1 100.0UF
R3 2 3 [1.2,.06]K
RC 3 6 [1.0,0.0]
RB 1 5 [100.0,5.0]
R2 2 1 [61.0,3.1]K
R1 1 0 [17.6,0.9]K
RE 4 7 [1.0,0.0]
R4 7 0 [300.0,15.0]
Rbc 5 6 [1.0e8,0.0]
Rbe 5 4 [1.0e8,0.0]
Rce 6 4 [1.0e8,0.0]
Q1 6 5 4 QNL
.op
.AC LIN 1 100 100hz
.model QNL NPN(BF=[80,12] CCS=2PF TF=0.3NS TR=6NS CJE=3PF
+ CJC=2PF VA=[50,7.5])
.END
```

***Figure 2-1D.*** SFA input file for single-stage amplifier. See Figure 2-1A for circuit diagram.

# Table 2-1I. One-Stage Amplifier; DC Data Only

SAS

DISCRIMINANT ANALYSIS  CLASSIFICATION SUMMARY FOR CALIBRATION DATA: WORK.FAULTS

GENERALIZED SQUARED DISTANCE FUNCTION  POSTERIOR PROBABILITY OF MEMBERSHIP IN EACH FAULT:

$$D^2_J(X) = (X-\bar{X}_J)' \; COV^{-1} \; (X-\bar{X}_J)$$

$$PR(J|X) = EXP(-.5\,D^2_J(X)) \; / \; SUM_K \; EXP(-.5\,D^2_K(X))$$

NUMBER OF OBSERVATIONS AND PERCENTS CLASSIFIED INTO FAULT:

| FROM FAULT: | NOMINAL | Q_HI | Q_LOW | RBC_SBCR | RBE_SBCR | RB_OPEN | RB_SBCRT | RCE_SBCR | RC_OPEN | RE_OPEN | TOTAL |
|---|---|---|---|---|---|---|---|---|---|---|---|
| NOMINAL | 108 | 45 | 0 | 0 | 0 | 0 | 47 | 0 | 0 | 0 | 200 |
| | 54.00 | 22.50 | 0.00 | 0.00 | 0.00 | 0.00 | 23.50 | 0.00 | 0.00 | 0.00 | 100.00 |
| Q_HI | 32 | 147 | 0 | 0 | 0 | 0 | 21 | 0 | 0 | 0 | 200 |
| | 16.00 | 73.50 | 0.00 | 0.00 | 0.00 | 0.00 | 10.50 | 0.00 | 0.00 | 0.00 | 100.00 |
| Q_LOW | 0 | 0 | 200 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 200 |
| | 0.00 | 0.00 | 100.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 100.00 |
| RBC_SBCR | 0 | 0 | 0 | 200 | 0 | 0 | 0 | 0 | 0 | 0 | 200 |
| | 0.00 | 0.00 | 0.00 | 100.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 100.00 |
| RBE_SBCR | 0 | 0 | 0 | 0 | 200 | 0 | 0 | 0 | 0 | 0 | 200 |
| | 0.00 | 0.00 | 0.00 | 0.00 | 100.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 100.00 |
| RB_OPEN | 0 | 0 | 0 | 0 | 0 | 125 | 0 | 0 | 0 | 74 | 200 |
| | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 62.50 | 0.00 | 0.00 | 0.00 | 37.00 | 100.00 |
| RB_SBCRT | 93 | 32 | 0 | 0 | 0 | 0 | 53 | 0 | 0 | 0 | 200 |
| | 46.50 | 16.00 | 0.00 | 0.00 | 0.00 | 0.00 | 26.50 | 0.00 | 0.00 | 0.00 | 100.00 |
| RCE_SBCR | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 200 | 0 | 0 | 200 |
| | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 100.00 | 0.00 | 0.00 | 100.00 |
| RC_OPEN | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 200 | 0 | 200 |
| | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 100.00 | 0.00 | 100.00 |
| RE_OPEN | 0 | 0 | 0 | 0 | 0 | 79 | 0 | 0 | 0 | 121 | 200 |
| | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 39.50 | 0.00 | 0.00 | 0.00 | 60.50 | 100.00 |
| TOTAL | 233 | 244 | 200 | 200 | 200 | 203 | 121 | 200 | 200 | 195 | 2000 |
| PERCENT | 11.65 | 12.20 | 10.00 | 10.00 | 10.00 | 10.15 | 6.05 | 10.00 | 10.00 | 9.75 | 100.00 |
| PRIORS | 0.1000 | 0.1000 | 0.1000 | 0.1000 | 0.1000 | 0.1000 | 0.1000 | 0.1000 | 0.1000 | 0.1000 | |

# Table 2-1J. One-Stage Amplifier; AC and DC Data

SAS

DISCRIMINANT ANALYSIS  CLASSIFICATION SUMMARY FOR CALIBRATION DATA  WORK.FAULTS

GENERALIZED SQUARED DISTANCE FUNCTION  POSTERIOR PROBABILITY OF MEMBERSHIP IN EACH FAULT1.

$$D^2_J(X) = (X-\bar{X}_J)' \, COV_J^{-1} \, (X-\bar{X}_J) + LN\,|COV_J| \qquad PR(J|X) = EXP(-.5\,D^2_J(X)) \,/\, SUM_K\, EXP(-.5\,D^2_K(X))$$

NUMBER OF OBSERVATIONS AND PERCENTS CLASSIFIED INTO FAULT1.

| FROM FAULT1 | NOMINAL | Q_HI | Q_LOW | RBC_SEOR | RBE_SEOR | RB_OPEN | RB_SECAT | RCE_SEOR | RC_OPEN | RE_OPEN | TOTAL |
|---|---|---|---|---|---|---|---|---|---|---|---|
| NOMINAL | 199 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 200 |
| | 99.50 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.50 | 0.00 | 0.00 | 0.00 | 100.00 |
| Q_HI | 1 | 199 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 200 |
| | 0.50 | 99.50 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 100.00 |
| Q_LOW | 0 | 0 | 200 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 200 |
| | 0.00 | 0.00 | 100.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 100.00 |
| RBC_SEOR | 0 | 0 | 0 | 200 | 0 | 0 | 0 | 0 | 0 | 0 | 200 |
| | 0.00 | 0.00 | 0.00 | 100.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 100.00 |
| RBE_SEOR | 0 | 0 | 0 | 0 | 200 | 0 | 0 | 0 | 0 | 0 | 200 |
| | 0.00 | 0.00 | 0.00 | 0.00 | 100.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 100.00 |
| RB_OPEN | 0 | 0 | 0 | 0 | 0 | 200 | 0 | 0 | 0 | 0 | 200 |
| | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 100.00 | 0.00 | 0.00 | 0.00 | 0.00 | 100.00 |
| RB_SECAT | 4 | 0 | 0 | 0 | 0 | 0 | 196 | 0 | 0 | 0 | 200 |
| | 2.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 98.00 | 0.00 | 0.00 | 0.00 | 100.00 |
| RCE_SEOR | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 200 | 0 | 0 | 200 |
| | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 100.00 | 0.00 | 0.00 | 100.00 |
| RC_OPEN | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 200 | 0 | 200 |
| | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 100.00 | 0.00 | 100.00 |
| RE_OPEN | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 200 | 200 |
| | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 100.00 | 100.00 |
| TOTAL | 204 | 198 | 200 | 200 | 200 | 200 | 197 | 200 | 200 | 200 | 2000 |
| PERCENT | 10.20 | 9.85 | 10.00 | 10.00 | 10.00 | 10.00 | 9.85 | 10.00 | 10.00 | 10.00 | 100.00 |
| PRIORS | 0.1000 | 0.1000 | 0.1000 | 0.1000 | 0.1000 | 0.1000 | 0.1000 | 0.1000 | 0.1000 | 0.1000 | |

## 2-1J. Study of BJT Parameter Sensitivities

In order to arrive at efficient models that account for variations in the behavior of BJTs, we performed a simple sensitivity analysis of a typical BJT's performance by varying key parameters in the SPICE Gummel-Poon BJT model. Random variation in key parameters during simulation would provide a nondeterministic model of the BJT that is used to build up the training sets for our multivariate studies.

In constructing the model, we began with a representative NPN transistor described by the following SPICE file:

```
BJT family of curves
.DC VCE 0 10.0V 0.2V IB 10UA 50UA 20UA
.print DC i(vic) i(vib) v(1)
ib 0 5
vib 1 5
Q4 3 1 0 QN2222
VIC 2 3
VCE 2 0
.MODEL QN2222 NPN(IS=1.9E-14 BF=150 VAF=100
+ IKF=.175 ISE=5E-11 NE=2.5 BR=7.5 VAR=6.38
+ IKR=0.012 ISC=1.9E-13 NC=1.2 RC=.4 XTB=1.5
+ CJE=26PF TF=.5E-9 CJC=11PF TR=30E-9)
.END
```

The above file constructs the I-V characteristics of a BJT under forward collector-emitter bias. In the sensitivity analysis, all QN2222 parameters specified in the above example were varied +20%; the effect of this variation on the I-V characteristics was then observed. Table 2-1K summarizes the parameter changes and their effect on the device operation. The table indicates that the parameters BF (forward bias beta) and NC (forward current emission coefficient) have much stronger effects on collector current than the other parameters. These effects can be observed in Figure 2-1E, where I-V characteristics have been plotted for nominal BF and NC values, and with these parameters increased by 20%. BF tends to vary the collector current at saturation, whereas NC effects collector current prior to saturation.

In view of the graphs of Figure 2-1E, it appears that a simple randomization of BF and NC will provide a good representation of variability in a bipolar transistor under normal forward bias conditions. The effect of beta variation holds even at much larger Vce values (exceeding 10V in this case). Clearly other parameters would exhibit more sensitivities at higher Vce (such as the early forward voltage VAF). However, changes in beta can easily mask such effects. Note that a drop

in beta (to near zero) can also be used to emulate a "dying" or "dead" transistor; i.e., a transistor without current gain. Of course, a faulty transistor can demonstrate any of a number of other characteristics, such as shifts in leakage currents, resistance, etc.

We must emphasize that the above conclusion has no basis in physics, nor does it reflect the multitude of process problems that can occur. However, from a circuit design and simulation point of view, variation in BF and NC does appear to cover a wide range in transistor I-V characteristics that will result in circuit performance variability.

**Table 2-1K. Effect of Variation in Gummel-Poon BJT Model Parameters**

| Param | Value nominal | Value +20% | Ic (mA) Vce=.1V nom | Ic (mA) Vce=.1V +20% | Ic (mA) Vce=.3V nom | Ic (mA) Vce=.3V +20% | |
|---|---|---|---|---|---|---|---|
| IS | 1.9e-14 | 2.3e-14 | 1.5 | 1.7 | 3.7 | 3.7 | |
| BF | 150 | 180 | 1.5 | 1.7 | 3.7 | 4.4 | X |
| VAF | 100 | 120 | 1.5 | 1.5 | 3.7 | 3.7 | |
| IKF | .175 | .210 | 1.5 | 1.5 | 3.7 | 3.7 | |
| ISE | 5.0e-14 | 6.0e-11 | 1.5 | 1.5 | 3.7 | 3.7 | |
| NE | 2.5 | 3.0 | 1.5 | 1.5 | 3.7 | 3.9 | |
| BR | 7.5 | 9.0 | 1.5 | 1.5 | 3.7 | 3.7 | |
| VAR | 6.38 | 7.65 | 1.5 | 1.5 | 3.7 | 3.8 | |
| IKR | 0.012 | 0.014 | 1.5 | 1.5 | 3.7 | 3.7 | |
| ISC | 1.9e-13 | 2.3e-13 | 1.5 | 1.4 | 3.7 | 3.7 | |
| NC | 1.2 | 1.4 | 1.5 | 2.5 | 3.7 | 3.7 | X |
| XTB | 1.5 | 1.8 | 1.5 | 1.5 | 3.7 | 3.7 | |

The symbol X indicates a significant effect

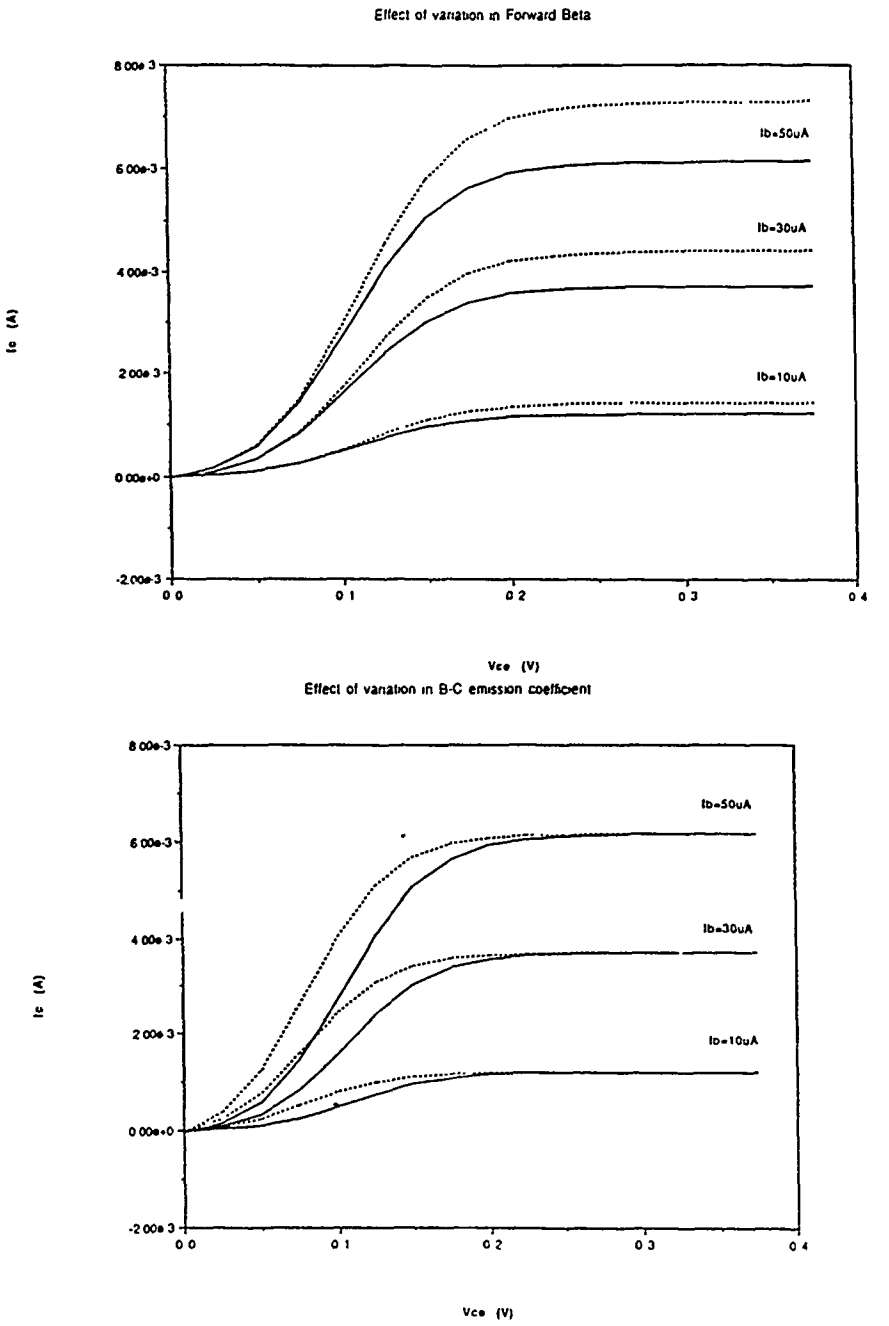



***Figure 2-1E.*** (Top) Simulated BJT I-V characteristics where BF was increased by 20% (dashed lines). (Bottom) Effect of increase in NC by 20% (dashed lines).

## 2.1K. Application of Plackett-Burman Experimental Designs

In our implementation, we used a 28 run Plackett-Burman design to reduce the number of model parameters from thirteen. The design layout is included in Figure 2-1F. In order to perform the experiment, one simulation of the circuit is performed for each row of the design. The values of the parameters are varied from the nominal by a fixed amount, in the direction indicated by the plus and minus signs in the first 13 columns of Figure 2-1F. For example, 11 of the 13 parameters were modified by 10%, and 2 variables were modified by 5%. In the first simulation run, the parameters 1, 3, 4, 5, 6, and 11 (parameters correspond to each row of Table 2-1J) were modified by +10% from the nominal, while the other parameters were decreased by 10% for parameters 2, 7, 8, 9, 10, and by 5% for parameters 12 and 13. Each row of Figure 2-1F corresponds to a simulation with parameters modified in a similar fashion.

The output of each simulation variable is recorded, in this case $v_1$, $v_3$ and $v_4$. For each variable, the values are added and subtracted down each column according to the pluses and minuses in Figure 2-1F, and the result is squared. Thus, the column operation is performed 27 times for each of the three voltage variables. For the first thirteen columns, this provides an estimate of the effect of the 13 parameters. For the remaining 14 columns, this provides an estimate of the standard deviation of the output variable. This result is compared to the estimate of the standard deviation of the variable which is obtained by averaging the squares of the effects in the remaining 14 columns. Specifically, a t-score is computed by dividing the (squared) effect of each parameter by the estimate of the standard deviation. This t-score is compared to the critical value of the t distribution with 14 degrees of freedom (14 d.f. because there are 14 estimates of the standard deviation).

This process is repeated for each output variable. Table 2-1J indicates the impact that each of the thirteen parameters has on each of the three output variables. Entries marked with a (**) denote high statistical significance (greater than 99%), while the (*) indicates marginally significant to significant effects (greater than 90%). The critical one-tailed t-value for 14 degrees of freedom is t(.90) = 1.345, while t(.99) = 2.624. Note that only a few parameters (in this case only R1, R2, R3, R5 and BF1) have highly significant impact on the simulation results, a few more (four) are marginally significant, and 4 parameters have no significant effect. Thus, by performing only 28 sim-

| Trial | Mean | x 1 | x 2 | x 3 | x 4 | x 5 | x 6 | x 7 | x 8 | x 9 | x 10 | x 11 | x 12 | x 13 | x 14 | x 15 | x 16 | x 17 | x 18 | x 19 | x 20 | x 21 | x 22 | x 23 | x 24 | x 25 | x 26 | x 27 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | + | + | - | + | + | + | + | - | - | - | - | + | - | - | - | + | - | - | + | + | + | - | + | - | + | + | - | + |
| 2 | + | + | + | - | + | + | + | - | - | - | - | - | + | + | - | - | + | - | - | - | + | + | + | + | - | + | + | - |
| 3 | + | - | + | + | + | + | + | - | - | - | + | - | - | - | + | - | - | + | - | + | - | + | - | + | + | - | + | + |
| 4 | + | - | - | - | + | - | + | + | + | + | - | - | + | - | + | - | - | - | + | + | - | + | + | + | - | + | - | + |
| 5 | + | - | - | - | + | + | - | + | + | + | + | - | - | - | - | + | + | - | - | + | + | - | - | + | + | + | + | - |
| 6 | + | - | - | - | - | + | + | + | + | + | - | + | - | + | - | - | - | + | - | - | + | + | + | - | + | - | + | + |
| 7 | + | + | + | + | - | - | - | + | - | + | - | - | + | - | - | + | - | + | - | + | - | + | + | - | + | + | + | - |
| 8 | + | + | + | + | - | - | - | + | + | - | + | - | - | + | - | - | - | - | + | + | + | - | + | + | - | - | + | + |
| 9 | + | + | + | + | - | - | - | - | + | + | - | + | - | - | + | - | + | - | - | - | + | + | - | + | + | + | - | + |
| 10 | + | - | + | - | - | - | + | - | - | + | + | + | - | + | - | + | + | - | + | + | - | + | + | + | + | - | - | - |
| 11 | + | - | - | + | + | - | - | + | - | - | - | + | + | + | + | - | + | + | - | + | + | - | + | + | + | - | - | - |
| 12 | + | + | - | - | - | + | - | - | + | - | + | - | + | - | + | + | - | + | + | - | + | + | + | + | + | - | - | - |
| 13 | + | - | - | + | - | + | - | - | - | + | + | - | + | + | + | - | + | - | + | - | - | - | + | - | + | + | + | + |
| 14 | + | + | - | - | - | - | + | + | - | - | + | + | - | - | + | + | + | + | - | - | - | - | + | + | - | + | + | + |
| 15 | + | - | + | - | + | - | - | - | + | - | - | + | + | + | - | + | - | + | + | - | - | - | - | + | + | + | + | + |
| 16 | + | - | - | + | - | - | + | - | + | - | + | - | + | + | - | + | + | + | - | + | + | + | - | - | - | + | - | + |
| 17 | + | + | - | - | + | - | - | - | - | + | + | + | - | + | + | - | - | + | + | + | + | + | - | - | - | + | + | - |
| 18 | + | - | + | - | - | + | - | + | - | - | - | + | + | - | + | + | + | - | + | + | + | + | - | - | - | - | + | + |
| 19 | + | + | + | - | + | - | + | + | - | + | + | - | + | + | + | + | - | - | - | - | + | - | - | + | - | - | + | - |
| 20 | + | - | + | + | + | + | - | + | + | - | + | + | - | + | + | + | - | - | - | - | - | + | + | - | + | - | - | - |
| 21 | + | + | - | + | - | + | + | - | + | + | - | + | + | + | + | + | - | - | - | + | - | - | - | + | - | - | + | - |
| 22 | + | + | - | + | + | + | - | + | - | + | - | - | - | + | - | + | + | + | + | - | - | + | - | + | - | - | - | + |
| 23 | + | + | + | - | - | + | + | + | + | - | - | - | - | + | + | - | + | + | + | + | - | - | - | - | + | + | - | - |
| 24 | + | - | + | + | + | - | + | - | + | + | - | - | - | - | + | + | + | + | + | - | + | - | + | - | - | - | + | - |
| 25 | + | + | - | + | + | - | + | + | + | - | + | + | + | - | - | - | + | - | + | - | - | + | - | - | + | - | + | - |
| 26 | + | + | + | - | + | + | - | - | + | + | + | + | + | - | - | - | + | + | - | + | - | - | + | - | - | - | - | + |
| 27 | + | - | + | + | - | + | + | + | - | + | + | + | + | - | - | - | - | + | + | - | + | - | - | + | - | + | - | - |
| 28 | + | - | - | - | - | - | - | - | - | - | - | - | - | - | - | - | - | - | - | - | - | - | - | - | - | - | - | - |

*Figure 2-1F.* 28 run Plackett-Burman design.

ulation runs, it is possible to reduce the number of parameters to be varied in future explorations of the simulation model from 13 to 9, and possibly down to 5. Of course, a larger number of parameters would require more simulations; therefore, the utility of the method may become questionable for larger circuits. In such cases, a closer look at dependencies between parameters, or on common factors (e.g., temperature, process parameters, etc.) would become more instructive in describing circuit variability.

For greater assurance, it is possible to repeat the experiment using different values for the increment or decrement from the nominal parameter values. For example, a parameter could be varied by 20% or 30%, rather than by the 10% used above. In the simulation model above, the outcomes of these further indicated even fewer significant parameters.

**Table 2-1J. Results of Plackett-Burman Experimental Screening for One-Stage Amplifier**

| Parameter | V1 | V3 | V4 |
|---|---|---|---|
| R1 10% | -10.23** | 11.24** | -12.25** |
| R2 10% | 12.63** | -12.08** | 13.49** |
| R3 10% | 1.076 | 5.534** | 0.463 |
| R5 10% | -1.308 | -4.647** | -1.436* |
| RB1 10% | 0.376 | -0.956 | 1.131 |
| RC1 10% | 0.718 | -1.400* | 1.542* |
| RE1 10% | -0.562 | 0.773 | -1.365* |
| RS1 10% | -0.214 | 1.594* | -0.925 |
| BF1 10% | -3.015** | 2.231* | -2.656** |
| BR1 10% | -0.466 | 0.512 | -1.318 |
| IS1 10% | 0.412 | -0.477 | 0.437 |
| VAF1 5% | -1.050 | 0.920 | -0.394 |
| VAR1 5% | -0.744 | 1.981* | -1.575* |

## 2-2. Differential Pair

### 2-2A. Discrimination Analysis

Figure 2-2A shows the differential pair circuit of our study, with the corresponding SFA file given in Figure 2-2B. This circuit was drawn from the SPICE3C1 documentation, and has become a popular simulator benchmark circuit. Transistors Q1 and Q2 make up the differential amplifier itself, with Q3 and Q4 serving as a current source for the pair. The fault analysis concentrates on transistor faults, the modeling of which should be self-explanatory from the circuit input file (Figure 2-2B). Note the use of SPICE subcircuit blocks in this example. Each subcircuit represents each transistor and its associated fault simulation components. Although all of the transistors are of the same type, separate subcircuit blocks must be supplied to account for the independent variability and faulting of each instance of the transistor in the circuit.

This example accounts for 8 fault types for each transistor. Given four transistors, we then have 4 times 8 or 32 fault classes, plus one additional class to represent nominal operation of the circuit. 20 or 200 points were generated for each fault class, with random gaussian perturbations in the circuit parameters (this results in a maximum total of 33 times 200, or 6600 points). Total computation time on a SUN 4/260 was about 2 hours. After a SAS DISCRIM run using 6 accessible DC voltages, we obtained the results given in Table 2-2A (data columns 1 through 4).

For 200 runs/fault, a comparison of columns 1 and 2 confirms the danger of using .aining set data for testing purposes (see Section 1.2B). Although the overall classification error only increases from 18% to 24%, drastic changes in classification error for specific faults occurs (e.g., RBE3_-SHORT, RC3_OPEN,RCE4_SHORT, RE1_OPEN). Likewise, the change in overall error in going from 200 to 20 points/fault is not very large; however, the change is greater for specific fault cases.

Most of the classification failures can be explained by considering the operation of the transistors, given the circuit's design. For example, all of the cases with transistor base resistances shorted could not be readily distinguished from the nominal cases. This is reasonable, given the forward bias of Q3 and Q4, which set the current through Q1 and Q2 of the differential pair. The high input resistances connected to the bases of Q1 and Q2 would further obscure drops in the base resistances of these devices. Likewise, shorting Q4's base and collector would have no effect since the design

already calls for this (neglecting the relatively small series base and collector resistances). The ambiguity between open-circuiting Q4's collector or emitter can be explained by the disruption of the reference current through Rbias, which would result in similar effects on the DC voltages at the nodes used in this analysis. Consequently, the faults RB_OPEN and RE_OPEN constitute an ambiguity set.

We extended the classification analysis by simply grouping all ambiguous classes with 200 points/fault (i.e., classes with greater than 2% error classification rate) into one class named "NOTNOMINAL." Using this classification, along with that of the other fault classes, we reran the SAS DISCRIM routine. Of the remaining fault classes except two, at least 98% of the Monte Carlo runs for each fault class were properly classified (in fact, we had 100% correct classification in most of the cases). Two fault classes, RB4_OPEN and RE4_OPEN, could not be distinguished from each other. This seems reasonable, since both faults effectively suppress the current source formed by the Q3-Q4 pair. In summary, once ambiguity sets are grouped, some degree of classification can be performed using a discrimination analysis.

### 2.2B. Note on the Assumption of Normality

The SAS DISCRIM analysis (see Part 1, Section 1.2B) assumes that the data in each class has normal variation. However, in view of the nonlinearities of the circuit simulation, not to mention data obtained from real IC fabrication lines, the assumption of normality does not hold true (although, in many cases, it may be a reasonable approximation). To avoid the assumption of normality, we have applied the SAS NEIGHBOR routine to the data of the differential pair. NEIGHBOR is a non-parametric discriminator that classifies objects into groups according to the *k-nearest-neighbor* rule (Section 1.5D). In our use of the command, a fault class for a given test voltage set was identified by determining the dominant class among the 33 nearest points to each test point. The value of "33 nearest neighbors" was chosen on an *ad hoc* basis since there are 33 fault classes. As the last two columns of Table 2-2A indicate, the classification power is very slightly degraded when using the nearest k neighbor method (compare columns 2 and 6). Another observation is that there is not much difference between the individual class or overall results when the original training set or new test data are used. Nevertheless, Table 2-2A points to the usefulness of a nonparametric method in performing classification. As mentioned in Part 1, however, such methods likely will require more computation time. (In fact, this is why the nearest k neighbor method was only applied with 20 samples/fault; 200 samples/fault would have required close to 24 hours of computation time on a MicroVax II, compared to about 20 min. for the discrimination analysis.)

## Table 2-2A. Differential Pair Fault Classification Results

*(all values expressed as % error)*

| | 200 points/fault | | 20 points/fault | | | |
|---|---|---|---|---|---|---|
| Fault Type | Discrim Analysis Training Set Data | Discrim Analysis New Test Data | Discrim Analysis Training Set Data | Discrim Analysis New Test Data | Nearest Neighbor Training Set Data | Nearest Neighbor New Test Data |
| Nominal | 73 | 96 | 70 | 85 | 95 | 90 |
| Q1_LOW | 0 | 0 | 0 | 0 | 0 | 0 |
| Q2_LOW | 0 | 0 | 0 | 0 | 0 | 0 |
| Q3_LOW | 18 | 0.5 | 0 | 5 | 0 | 5 |
| Q4_LOW | 18 | 13 | 15 | 35 | 10 | 30 |
| RB1_OPEN | 0 | 9 | 0 | 0 | 20 | 30 |
| RB1_SHORT | 69 | 79 | 45 | 60 | 55 | 60 |
| RB2_OPEN | 0 | 5 | 0 | 0 | 30 | 30 |
| RB2_SHORT | 54 | 74 | 50 | 95 | 85 | 85 |
| RB3_OPEN | 18 | 0 | 15 | 20 | 0 | 0 |
| RB3_SHORT | 61 | 46 | 55 | 50 | 85 | 75 |
| RB4_OPEN | 44 | 61 | 15 | 50 | 10 | 5 |
| RB4_SHORT | 42 | 45 | 50 | 70 | 60 | 55 |
| RBC1_SHORT | 2 | 9 | 5 | 25 | 10 | 5 |
| RBC2_SHORT | 4 | 12 | 0 | 0 | 5 | 0 |
| RBC3_SHORT | 0 | 0 | 0 | 0 | 0 | 0 |
| RBC4_SHORT | 91 | 98 | 45 | 80 | 80 | 80 |
| RBE1_SHORT | 0 | 0 | 0 | 0 | 0 | 0 |
| RBE2_SHORT | 0 | 0 | 0 | 0 | 0 | 0 |
| RBE3_SHORT | 1 | 61 | 0 | 0 | 0 | 0 |
| RBE4_SHORT | 0 | 39 | 0 | 0 | 35 | 35 |
| RC1_OPEN | 0 | 0 | 0 | 0 | 0 | 0 |
| RC2_OPEN | 0 | 0 | 0 | 0 | 0 | 0 |
| RC3_OPEN | 0 | 38 | 0 | 0 | 45 | 40 |
| RC4_OPEN | 0 | 0 | 0 | 0 | 0 | 0 |
| RCE1_SHORT | 0 | 10 | 0 | 0 | 30 | 25 |
| RCE2_SHORT | 0 | 9 | 0 | 0 | 10 | 15 |
| RCE3_SHORT | 0 | 0 | 0 | 0 | 0 | 0 |
| RCE4_SHORT | 83 | 0 | 0 | 0 | 0 | 0 |
| RE1_OPEN | 0 | 24 | 0 | 0 | 10 | 15 |
| RE2_OPEN | 0 | 18 | 0 | 0 | 5 | 35 |
| RE3_OPEN | 5 | 0 | 95 | 90 | 0 | 0 |
| RE4_OPEN | 27 | 33 | 50 | 40 | 75 | 70 |
| Total APER | 18 | 23 | 15 | 21 | 23 | 24 |

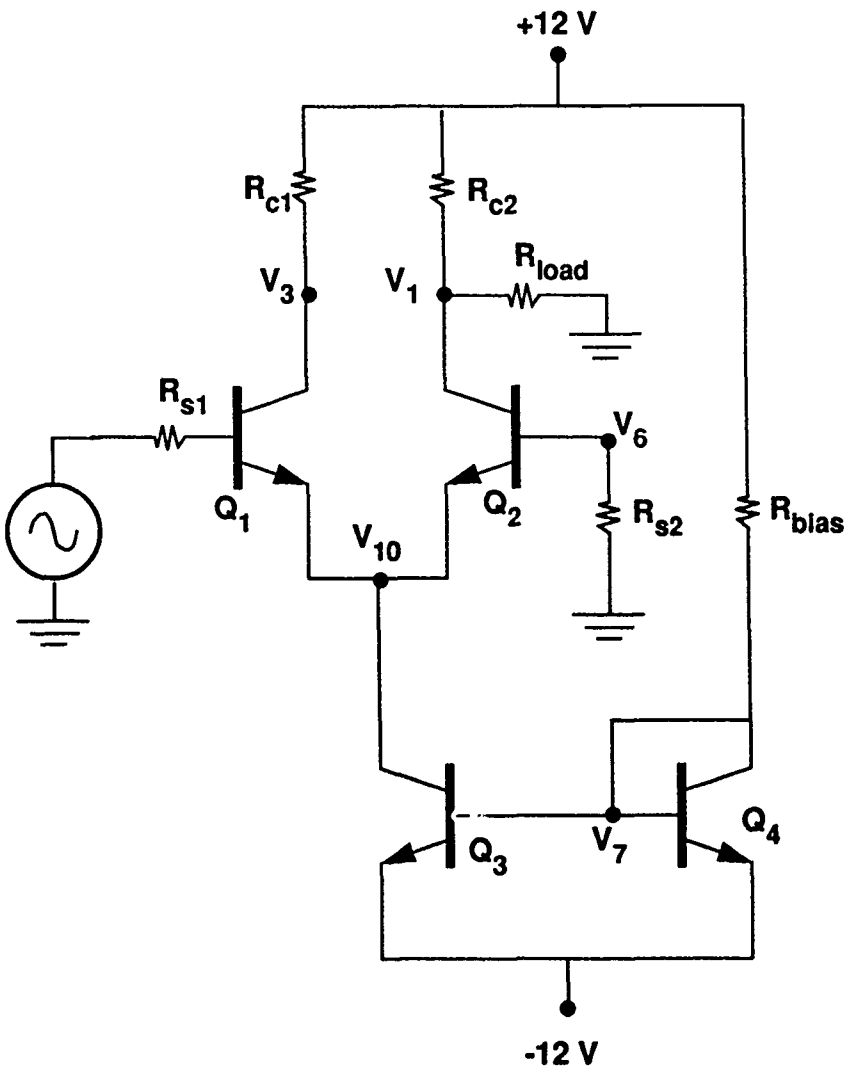



***Figure 2-2A.*** Differential pair circuit. The SFA file is shown in Figure 2-2B (next page).

```
Differential Pair w/ active current source
* Nominal circuit:
#GOOD 200
*
* Q1 faults:
#RB1 1.0e6 200 RB1_open
#RB1 1.0 200 RB1_short
#RC1 1.0e6 200 RC1_open
#RE1 1.0e6 200 RE1_open
#RBC1 0.01 200 RBC1_short
#RBE1 0.01 200 RBE1_short
#RCE1 0.01 200 RCE1_short
#QNL1 10 200 Q1_low BF
*
* Q2 faults:
#RB2 1.0e6 200 RB2_open
#RB2 1.0 200 RB2_short
#RC2 1.0e6 200 RC2_open
#RE2 1.0e6 200 RE2_open
#RBC2 0.01 200 RBC2_short
#RBE2 0.01 200 RBE2_short
#RCE2 0.01 200 RCE2_short
#QNL2 10 200 Q2_low BF
*
* Q3 faults:
#RB3 1.0e6 200 RB3_open
#RB3 1.0 200 RB3_short
#RC3 1.0e6 200 RC3_open
#RE3 1.0e6 200 RE3_open
#RBC3 0.01 200 RBC3_short
#RBE3 0.01 200 RBE3_short
#RCE3 0.01 200 RCE3_short
#QNL3 10 200 Q3_low BF
*
* Q4 faults:
#RB4 1.0e6 200 RB4_open
#RB4 1.0 200 RB4_short
#RC4 1.0e6 200 RC4_open
#RE4 1.0e6 200 RE4_open
#RBC4 0.01 200 RBC4_short
#RBE4 0.01 200 RBE4_short
#RCE4 0.01 200 RCE4_short
#QNL4 10 200 Q4_low BF
*
%dc 1 2 3 6 7 10
*
.SUBCKT QT1 1 2 3
RB1 2 12 [100.0,0]
RC1 1 11 [1.0,0]
RE1 3 13 [1.0,0]
RBC1 11 12 [10000.0,0]K
RBE1 12 13 [10000.0,0]K
RCE1 11 13 [10000.0,0]K
Q1 11 12 13 QNL1
* Q1 1 2 3 QNL1
.MODEL QNL1 NPN (BF=[80,12] CCS=2PF
+TF=0.3NS TR=6NS CJE=3PF CJC=2PF VA=[50,8])
.ENDS QT1
*
.SUBCKT QT2 1 2 3
RB2 2 12 [100.0,0]
RC2 1 11 [1.0,0]
RE2 3 13 [1.0,0]
RBC2 11 12 [10000.0,0]K
RBE2 12 13 [10000.0,0]K
RCE2 11 13 [10000.0,0]K
Q2 11 12 13 QNL2
* Q2 1 2 3 QNL2
.MODEL QNL2 NPN (BF=[80,12] CCS=2PF
+ TF=0.3NS TR=6NS CJE=3PF
+ CJC=2PF VA=[50,8])
.ENDS QT2
*
.SUBCKT QT3 1 2 3
RB3 2 12 [100.0,0]
RC3 1 11 [1.0,0]
RE3 3 13 [1.0,0]
RBC3 11 12 [10000.0,0]K
RBE3 12 13 [10000.0,0]K
RCE3 11 13 [10000.0,0]K
Q3 11 12 13 QNL3
* Q3 1 2 3 QNL3
.MODEL QNL3 NPN (BF=[80,12] CCS=2PF
+ TF=0.3NS TR=6NS CJE=3PF
+ CJC=2PF VA=[50,8])
.ENDS QT3
*
.SUBCKT QT4 1 2 3
RB4 2 12 [100.0,0]
RC4 1 11 [1.0,0]
RE4 3 13 [1.0,0]
RBC4 11 12 [10000.0,0]K
RBE4 12 13 [10000.0,0]K
RCE4 11 13 [10000.0,0]K
Q4 11 12 13 QNL4
* Q4 1 2 3 QNL4
.MODEL QNL4 NPN (BF=[80,12] CCS=2PF
+ TF=0.3NS TR=6NS CJE=3PF
+ CJC=2PF VA=[50,8])
.ENDS QT4
*
* Main Circuit:
* .DC VIN -0.25 .25 0.005
.op
RC1 3 4 [10,.5]K
XQ2 1 6 10 QT2
XQ1 3 2 10 QT1
RC2 1 4 [10,.5]K
RBIAS 4 7 [20,1]K
VCC 4 0 12
VIN 5 0 0.0
VEE 8 0 -12.0
XQ3 10 7 8 QT3
XQ4 7 7 8 QT4
R5 6 0 [1,.05]K
RS1 5 2 [1,.05]K
.END
```

***Figure 2-2B.*** SFA Input file for differential pari circuit.

## 2.3. Operational Amplifier -- Go/No-Go Testing

### 2.3A. Circuit Description

The simulation of an operational amplifier proved to be a severe challenge for this study. To conserve computation time, we chose to use a nonlinear behavioral model for an op-amp, rather than a full transistor-level model. Figure2-3A shows the model used.

The model of Figure 2-3A is based on the work published by Boyle *et. al.* [1]. This model (with variations) appears in several references, including the *Raytheon RLA* guide [14] and the *Intusoft* SPICE guide [17]. The Intusoft version, which corresponds to a UA741, has been adapted for this study. It accounts for the real circuit's behavior at its input by use of the differential pair made up of Q1 and Q2. Current sources GA and GCM model the differential and common mode gain, respectively, of the device. Current source GB represents the gain of an intermediate stage in the real circuit, while transistors Q4 and Q5 with diodes D6 and D7 model the output stage. Diodes D2 and D3 model current limiting, while diodes D4 and D5 model voltage limiting.

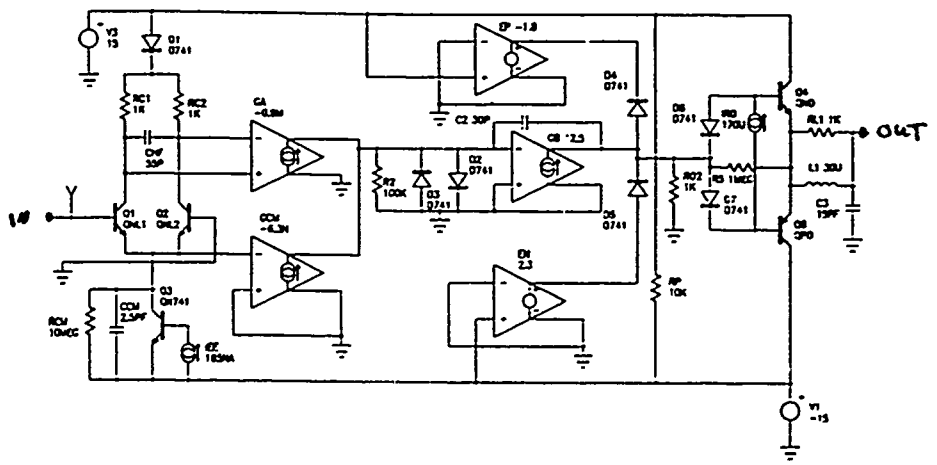



***Figure 2-3A.*** Behavioral model of op-amp used in this study (based on ref.1,14 and 17).

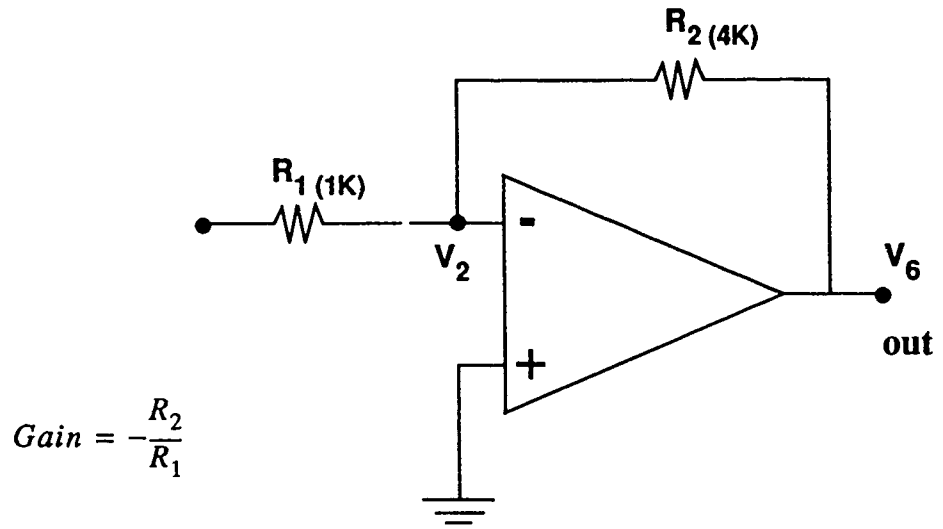



***Figure 2-3B.*** Inverting amplifier configuration. The op-amp is represented by the behavioral model of Figure 2-3A.

The model of Figure 2-3A, and its variants (including that described in the Boyle *et. al.* reference [1] and in the RLA guide [14]) proved difficult to simulate. For example, Berkeley *SPICE3C1*, *IG-SPICE (2G6)* and *Pspice (2G6)* either had problems converging during the DC operating point analysis, or produced radically different answers without any warnings! These problems occurred regardless of the setting of the SPICE OPTIONS parameters, in particular the ITL1 (for iteration count) and ITL6 (for source stepping) parameters. Likewise, setting initial conditions (using NODESET) had no effect. In the end, the model of Figure 2-3A was chosen with the diodes D4 and D5, as well as the voltage sources EP and EN removed. This does not pose a severe limitation for our analysis since voltage limiting does not enter into the faults studied. Nevertheless, even this reduced configuration led to occasional convergence problems during the Monte Carlo runs.

The model was placed in a simple inverting amplifier configuration, as shown in Figure 2-3B. The gain of this circuit was set to -4 using the indicated combination of $R_1$ and $R_2$. An arbitrarily statistical model of the "nominal" op-amp was generated by varying the component parameters of the behavioral model by $\pm$ 5%. We do not claim that this represents the true statistical variation of any op-amp production run; rather, the procedure simply provides nominal variation for the pur-

poses of our study. Ten faults were arbitrarily introduced within and outside the op-amp (refer to Figure 2-3C for the SFA input file). Note that the analysis covered the DC voltages at nodes 2 and 6, as well as the complex AC voltages at these nodes for four frequencies between 10 Hz and 100KHz. Thus, a total of 18 values were obtained for each Monte Carlo iteration. Two hundred Monte Carlo iterations were performed for the nominal runs, while only one random iteration was performed for each of the 10 fault types.

Two methods were used to detect circuit failure during Go/No-Go testing. The first method made use of the SAS DISCRIM routine where all nominal runs were grouped into one class, and all faults were grouped into another, but single, class. The second method made use of multivariate hypothesis testing.

### 2.3B. Fault Analysis

***2-Class Discrimination:***

In this method, all 11 fault types were grouped into one class, then the discriminator was trained (by running DISCRIM) using the nominal and faulted data as the two classes. Reapplying the training set data for testing the discriminator, the following results were obtained:

Using "within class" covariance matrices (@ 99% threshold); all 18 variables:

| | |
|---|---|
| nominal: | all 200 nominal runs properly classified. |
| faults: | 8 classified as faulted; 3 misclassified as nominal. |

Using "pooled" covariance matrices; all 18 variables:

| | |
|---|---|
| nominal: | all 200 nominal runs properly classified. |
| faults: | 6 classified as faulted; 5 misclassified as nominal. |

Using only the 2 DC variables (at V2 and V6); 99% threshold:

| | |
|---|---|
| nominal: | 97% of 200 nominal properly classified; |
| faults: | 4 faults properly classified; 7 misclassified |

We see that exclusion of the AC data degrades the fault classification. Nevertheless, many of the faults in these simulations could not be distinguished from the nominal runs. Note that some (if not all) of the misclassification of the nominal runs in the DC-only test was due to the presence of outliers that resulted from SPICE convergence errors.

***Multivariate Hypothesis Test:***

This procedure makes use of the methodology described in Section 1.3. For the present example, the following procedure was followed:

1) SAS stepwise discrimination STEPDISC routine first eliminated variables (i.e., voltages) that did not aid in the separation of the nominal run class from the fault class defined by grouping all faults into one fault class. For the current problem, 5 voltages remained at a significance level of 0.05 (see the SAS manual).
2) The data were transferred to the an IBM-compatible PC. The program MathCAD [21] was used to compute the means and covariance matrix for the nominal case.
3) Using MathCAD, eq. 1-12 was computed for each fault.
4) The result was compared with the chi-square value for 5 degrees of freedom (corresponding to 5 voltages). At an $\alpha$ of 0.05, this is 11.07.

Following the above procedure, 4 of the 11 faults incorrectly passed the test (i.e., were incorrectly identified as "good"). A random sample of 11 nominal points was also tested. All of these points were correctly classified as nominal.

After we removed the outliers from the complete set of nominal runs (there were 7 outliers, all presumably due to SPICE convergence errors), we tested all the remaining nominal points. 182 out of 193 nominal points (94%) were correctly classified. Re-running the hypothesis testing on the faults, 8 out of 11 faults were correctly classified as "not nominal." The remaining three faults could not be distinguished from the nominal runs. Note that this result is identical to the DISCRIM analysis where the within-class covariances were used (discussed above).

Although DISCRIM may provide a quick and easy method for performing multivariate Go/No-Go testing, *it is also dangerous in this application.* During training for the collective fault class, a collection of faults may arise that have an effective mean almost identical to that of the nominal runs (i.e., the centroids of the two classes are the same.) In this case, the two classes become difficult to distinguish. We therefore would recommend the multivariate hypothesis testing with the straight-forward use of eq. 1-12.

*Figure 2-3C.* SFA input file for inverting amplifier using nonlinear op-amp behavioral model.

```
741 OP AMP Macro Model
* Based Boyle et. al.and Intusoft models
* Voltage saturation diodes removed
* to eliminate convergence problems
* B. Epstein  6/90
* Nominal circuit:
#GOOD 200
*
* Arbitrary faults:
#QNI1 10 1 QNI1_low BF
#QNI2 10 1 QNI2_low BF
#RC1 1000.0 1 RC1_open
#RC2 1000.0 1 RC2_open
#RP  0.01 1 RP_low
#R2 0.01 1 R2_low
#QPO 10.0 1 QPO_low BF
#RO2 20.0 1 RO2_low
#R1M 0.001 1 R1m_short
#R2M 0.001 1 R2m_short
#R2M 1000.0 1 R2m_open
*
%dc 2 6
%ac 2 6
```

```
.SUBCKT UA741 2   3   6   7  4
*             (-) (+) OUT  V+ V-
QNI1 10 2 13 QNI1
QNI2 12 3 13 QNI2
.MODEL QNI1 NPN (NF=1.5 BF=[111.0,10.0]
* IS=8E-16 CJE=3PF)
.MODEL QNI2 NPN (NF=1.5 BF=[144.0,10.0]
* IS=8.3E-16 CJE=3PF)
Q3 13 14 4 QN741
IEE 4 14 185NA
CCM 13 4 2.5PF
RCM 13 4 10MEG
RC1 11 10 [1.0,0.05]K
RC2 11 12 [1.0,0.05]K
CHF 10 12 55PF
D1 7 11 D741
RP 7 4 [10.0,0.5]K
GA 0 15 12 10 [0.9,0.05]MMHO
GCM 0 15 13 0 [6.3,0.03]NMHO
R2 15 0 [100.0,5.0]K
D2 15 0 D741 OFF
D3 0 15 D741 OFF
C2 15 16 30PF
GB 16 0 15 0 [12.5,1.0]
* D4 16 37 D741
* D5 34 16 D741
* RDD4 37 16 10K
* RDD5 34 16 10K
* VCP 37 0 13.0
* VCN 34 0 -13.0
RO2 16 0 [1000.0,50.0]
D6 19 16 D741 OFF
D7 16 20 D741 OFF
IRO 20 19 [170.0,8.0]UA
RRO 16 21 1MEG
Q4 7 19 21 QNO
Q5 4 20 21 QPO
.MODEL QNO NPN (BF=[150.0,7.5] IS=1E-14)
.MODEL QPO NPN (BF=[150.0,7.5] IS=1E-14)
L1 21 6 30UH
RL1 21 6 [1.0,0.05]K
.MODEL D741 D (CJO=3PF)
.MODEL QN741 NPN (IS=8E-16)
.ENDS
* Main circuit:
R1M 1 2 [1.0,0.05]k
R2M 2 6 [4.0,0.2]k
VIN 1 0 DC 2 AC 0.5
VP 7 0 15V
VN 4 0 -15V
X1 2 0 6 7 4 UA741
.op
.AC LIN 2 10 100K
.OPTIONS ITL1=500 ITL6=2
.end
```

## 2.4. Elliptical Filter

### 2.4A. Circuit Description

To further investigate Go/No-Go testing, we applied the above techniques to a larger circuit representing a fifth-order elliptical filter. This circuit was from the *Intusoft* SPICE manual, and makes use of a linearized version of the op-amp circuit in Figure 2-3A. The filter has three op-amps (see Figure 2-4A). An important characteristic of the circuit is its rapid gain roll-off at about 1 kHz. Variations in this roll-off help to discriminate faults.

Figure 2-4B shows the SFA input file for this example. Three nodes were selected for both DC and AC measurements at 5 frequencies ranging from 1 kHz to 5 kHz. Thus, a total of 33 voltages (3 DC + 2 X 15 -- don't forget real and imaginary parts) were obtained for each nominal iteration and fault. Random variation in each of the op-amps' component parameters (std. dv. 5%, normal), as well as nominal 5% variation in the interconnecting component parameters, defined the nominal statistics of the overall circuit. 22 arbitrary catastrophic faults were then introduced to the circuit components that interconnected the op-amps (see the list at the beginning of the SFA file in Figure 2-4B). Note that the capacitors were "shorted" by use of the resistor in parallel with each capacitor.

### 2.4B. Fault Analysis

Once again, we used the two methods discussed in the previous section for performing Go/No-go testing.

***2-Class Discrimination :***

Using only 2 DC voltages, no nominal cases were properly classified as nominal. However, using all 33 variables, 98% of the nominal cases were properly classified as nominal. None of the faulted cases were classified as nominal; i.e., *all* 22 faults were properly classified as faults!

***Multivariate Hypothesis Test:***

Following the procedure described above, the original 33 variables were reduced to 25 variables through the SAS STEPDISC procedure, then we computed the covariance matrix of the nominal runs. All 22 faults were correctly recognized as faults. Likewise, a sample of 40 random nominals was completely classified as nominal.

***Full Discrimination:***

We investigated a straight forward discrimination analysis, where each of the 22 fault types and the nominal cases represented a total of 23 classes. Applying the SAS DISCRIM routine to 20 Monte Carlo iterations for each of the 22 fault classes, along with 200 iterations for the nominal class, resulted in 100% accurate classification of all 22 fault classes (the training set data was used for the test). In other words, *the discriminator was able to detect and classify all 22 faults with 100% accuracy!* 98% of all nominal runs were classified as nominal.

The elliptical filter yielded quite impressive screening results, especially in view of the circuit's size and complexity. The success of the hypothesis testing and discrimination analysis is likely due to the strong dependence of the circuit's frequency roll-off on component values. This points to the importance of a good test strategy when performing Go/No-go testing and fault classification.

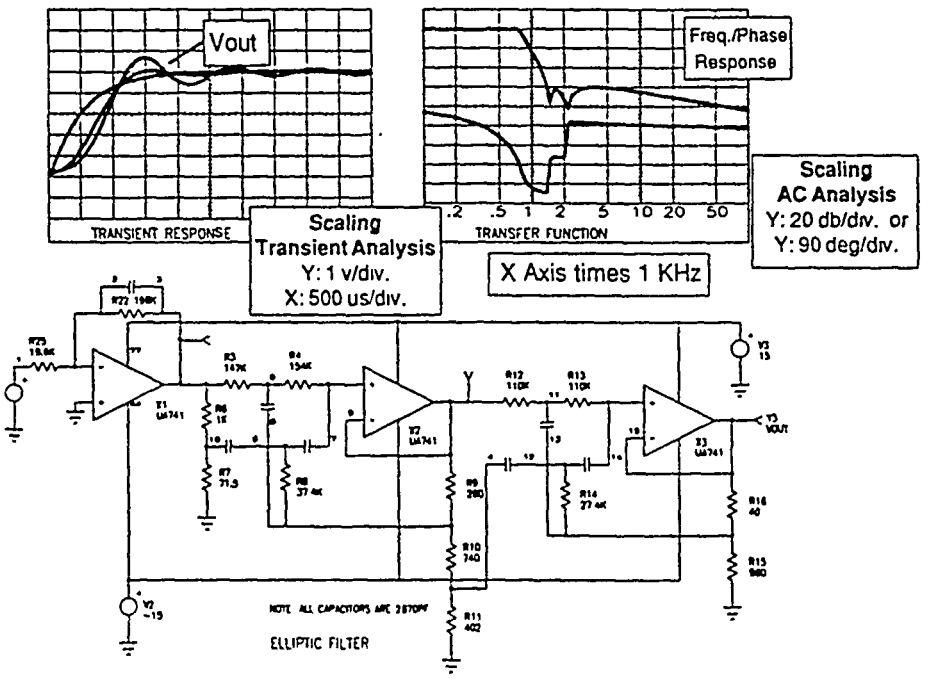



***Figure 2-4A.*** Fifth-order elliptical filter (from *Intusoft* manual, ref. 17).

```
Linear 5th order elliptical filter
* (from Intusoft)
#GOOD 200
#CM1 0.001 1 C1_low
#RF1 0.1 1 C1_short
#CM2 0.001 1 C2_low
#RF2 0.1 1 C2_short
#CM3 0.001 1 C3_low
#RF3 0.1 1 C3_short
#CM4 0.001 1 C4_low
#RF4 0.1 1 C4_short
#CM5 0.001 1 C5_low
#RF5 0.1 1 C6_short
#CM6 0.001 1 C7_low
#RF6 0.1 1 C7_short
#CM7 0.001 1 C8_low
#RF7 0.1 1 C8_short
#RM25 0.01 1 R25_low
#RM25 10000.0 1 R25_hi
#RM3 0.01 1 R3_low
#RM3 10000.0 1 R3_hi
#RM8 0.01 1 R8_low
#RM8 10000.0 1 R8_hi
#RM12 0.01 1 R12_low
#RM12 10000.0 1 R12_hi
*
%POLAR
%dc 3 9 15
%ac 3 9 15
*
.SUBCKT X1UA741 2  3  6   7   4
*               - IN + OUT VCC VEE
RP 4 7 [10.0,0.5]K
RXX 4 0 [10.0,0.5]MEG
IBP 3 0 [80.0,4.0]NA
RIP 3 0 [10.0,0.5]MEG
CIP 3 0 [1.4,0.07]PF
IBN 2 0 [100.0,5.0]NA
RIN 2 0 [10.0,0.5]MEG
CIN 2 0 [1.4,0.07]PF
VOFST 2 10 [1.0,0.05]MV
RID 10 3 [200.0,10.0]K
EA 11 0 10 3 1
R1 11 12 [5.0,.25]K
R2 12 13 [50,2.5]K
C1 12 0 [13.0,.65]PF
GA 0 14 0 13 [2700.0,135]
C2 13 14 [2.7,.14]PF
RO 14 0 [75.0,3.75]
L 14 6 [30.0,1.5]UHY
RL 14 6 [1000,50.0]
CL 6 0 [3.0,.15]PF
.ENDS X1UA741
```

```
.SUBCKT X2UA741 2  3  6   7   4
*               - IN + OUT VCC VEE
RP 4 7 [10.0,0.5]K
RXX 4 0 [10.0,0.5]MEG
IBP 3 0 [80.0,4.0]NA
RIP 3 0 [10.0,0.5]MEG
CIP 3 0 [1.4,0.07]PF
IBN 2 0 [100.0,5.0]NA
RIN 2 0 [10.0,0.5]MEG
CIN 2 0 [1.4,0.07]PF
VOFST 2 10 [1.0,0.05]MV
RID 10 3 [200.0,10.0]K
EA 11 0 10 3 1
R1 11 12 [5.0,.25]K
R2 12 13 [50,2.5]K
C1 12 0 [13.0,.65]PF
GA 0 14 0 13 [2700.0,135]
C2 13 14 [2.7,.14]PF
RO 14 0 [75.0,3.75]
L 14 6 [30.0,1.5]UHY
RL 14 6 [1000,50.0]
CL 6 0 [3.0,.15]PF
.ENDS X2UA741
*
.SUBCKT X3UA741 2  3  6   7   4
*               - IN + OUT VCC VEE
RP 4 7 [10.0,0.5]K
RXX 4 0 [10.0,0.5]MEG
IBP 3 0 [80.0,4.0]NA
RIP 3 0 [10.0,0.5]MEG
CIP 3 0 [1.4,0.07]PF
IBN 2 0 [100.0,5.0]NA
RIN 2 0 [10.0,0.5]MEG
CIN 2 0 [1.4,0.07]PF
VOFST 2 10 [1.0,0.05]MV
RID 10 3 [200.0,10.0]K
EA 11 0 10 3 1
R1 11 12 [5.0,.25]K
R2 12 13 [50,2.5]K
C1 12 0 [13.0,.65]PF
GA 0 14 0 13 [2700.0,135]
C2 13 14 [2.7,.14]PF
RO 14 0 [75.0,3.75]
L 14 6 [30.0,1.5]UHY
RL 14 6 [1000,50.0]
CL 6 0 [3.0,.15]PF
.ENDS X3UA741
*
```

***Figure 2-4B.*** SFA input for elliptical filter.

```
* Capacitor block:
.SUBCKT CNET 1 2 3 4 5 6 7 8 9 10 11 12 13 14
CM1 1 2 [2.6667,.13]NF
CM2 3 4 [2.6667,.13]NF
CM3 5 6 [2.6667,.13]NF
CM4 7 8 [2.6667,.13]NF
CM5 9 10 [2.6667,.13]NF
CM6 11 12 [2.6667,.13]NF
CM7 13 14 [2.6667,.13]NF
.ENDS CNET
* Capacitor fault block:
.SUBCKT RNET 1 2 3 4 5 6 7 8 9 10 11 12 13 14
RF1 1 2 [1.0e8,0.0]
RF2 3 4 [1.0e8,0.0]
RF3 5 6 [1.0e8,0.0]
RF4 7 8 [1.0e8,0.0]
RF5 9 10 [1.0e8,0.0]
RF6 11 12 [1.0e8,0.0]
RF7 13 14 [1.0e8,0.0]
.ENDS RNET
*
* MAIN CIRCUIT
X1 2 0 3 77 44 X1UA741
RM25 1 2 [19.6,1.0]K
RM22 2 3 [196.0,9.8]K
X2 9 7 9 77 44 X2UA741
RM3 3 5 [147.0,7.4]K
RM4 5 7 [154.0,7.7]K
RM6 3 10 [1.0,0.05]K
RM7 10 0 [71.5,3.6]
RM8 6 8 [37.4,1.9]K
X3 15 14 15 77 44 X3UA741
RM12 9 11 [110.0,5.0]K
RM13 11 14 [110.0,5.0]K
RM9 9 8 [260.0,13,0]
RM10 8 4 [740,37.0]
RM14 12 13 [27.4,1.4]K
RM16 15 13 [40.0,2.0]
RM11 4 0 [402.0,20.0]
RM15 13 0 [960.0,48.0]
V2 44 0 -15
V3 77 0 15
X4 2 3 10 6 5 8 6 7 4 12 13 11 12 14 CNET
X5 2 3 10 6 5 8 6 7 4 12 13 11 12 14 RNET
VIN 1 0 AC 1 DC 1
.op
.AC LIN 3 1K 5K
.print DC V(1) V(3) V(5) V(7)
.print AC VM(1) VM(3) VM(5) VM(7)
.END
```

***Figure 2-4B.*** SFA input for elliptical filter (continued).

## 2.5. Comparator Circuit

### 2.5A. Circuit description

A simple comparator circuit was simulated using SPICE in the SFA system. The circuit is shown in Figure 2-5A[2]. The configuration of the circuit is that of an inverting comparator with built-in hysteresis. As the input voltage ($E_{in}$) is ramped up, the comparator swings into negative saturation. However, during the downward ramp, positive saturation is attained.

The SFA input file is given in Figure 2-5B. The full 741 Op Amp circuit transistor-level SPICE description was used for the simulations (Figures 2-5B and 2-5C). Even though this model has 23 transistors, the simulations had better convergence properties than the simpler macromodels used in our previous examples. Figure 2-5D shows the response of the output of the comparator as the input voltage is ramped up from -15 V to +15 V, then down from +15 V to -15V. As expected, saturation occurs at about -15 V and +15V, which are the supply voltages to the op-amp. The analysis was performed using SPICE's DC computation mode (not transient analysis). The dependent voltage source controlled the direction of the ramp (examine the input file of Figure 2-5B).

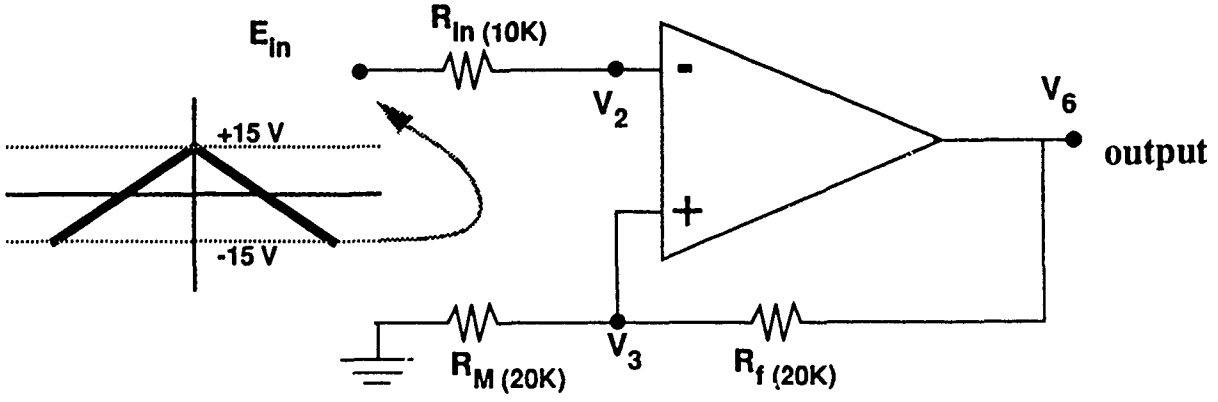



***Figure 2-5A.*** Noninverting comparator with hysteresis. A zero reference voltage is applied to resistor $R_M$ by grounding this resistor.

```
Comparator using full 741 op amp circuit
* B. Epstein  7/90
#GOOD 200
#Rin 10000.0 100 Rin_OPEN
#Rin 0.0001 100 Rin_LOW
#RM1 200000.0 100 R1_OPEN
#RM1 0.0001 100 R1_LOW
#Rf 200000.0 100 RF_OPEN
#Rf 0.0001 100 RF_LOW
%dc 6
*
.SUBCKT UA741 1   2  24   27 26
*             (-) (+) OUT  V+ V-
R1 10 26 [1.0,.05]K
R2 9 26 [50.0,2.5]K
R3 11 26 [1.0,0.05]K
R4 12 26 [3.0,0.15]K
R5 15 17 [39.0,2.0]K
R6 21 20 [40.0,2.0]K
R7 14 26 [50.0,2.5]K
R8 18 26 [50.0,2.5]
R9 24 25 [25.0,1.25]
R10 23 24 [50.0,2.0]
R11 13 26 [50.0,2.0]K
COMP 22 8 30PF
Q1 3 2 4 QNL
Q2 3 1 5 QNL
Q3 7 6 4 QPL
Q4 8 6 5 QPL
Q5 7 9 10 QNL
Q6 8 9 11 QNL
Q7 27 7 9 QNL
Q8 6 15 12 QNL
Q9 15 15 26 QNL
Q10 3 3 27 QPL
Q11 6 3 27 QPL
Q12 17 17 27 QPL
Q13 8 13 26 QNL
Q14 22 17 27 QPL
Q15 22 22 21 QNL
Q16 22 21 20 QNL
Q17 13 13 26 QNL
Q18 27 8 14 QNL
Q19 20 14 18 QNL
Q20 22 23 24 QNL
Q21 13 25 24 QPL
Q22 27 22 23 QNL
Q23 26 20 25 QPL
.MODEL QNL NPN(BF=[80.0,8.0] RB=[100.0,5.0] CCS=2PF TF=0.2NS TR=6NS CJE=3PF
+  CJC=2PF VA=[50.0,5.0])
.MODEL QPL PNP(BF=[10.0,1.0] RB=[20.0,1.0] TF=1NS TR=20NS CJE=6PF CJC=4PF
+  VA=[50.0,5.0])
.ENDS
*
* MAIN CIRCUIT
VCC 7 0 15.0
VEE 4 0 -15.0
VIN 22 0 DC
B1 12 0 V=15.0 - 2.0*abs(v(22))
Rin 12 2 [10.0,.5]K
RM1 3 0 [20.0,1.0]K
Rf 3 6 [20.0,1.0]K
X1 2 3 6 7 4 UA741
.dc VIN -15.0 15.0 1.0
.END
```

***Figure 2-5B.*** Comparator circuit.

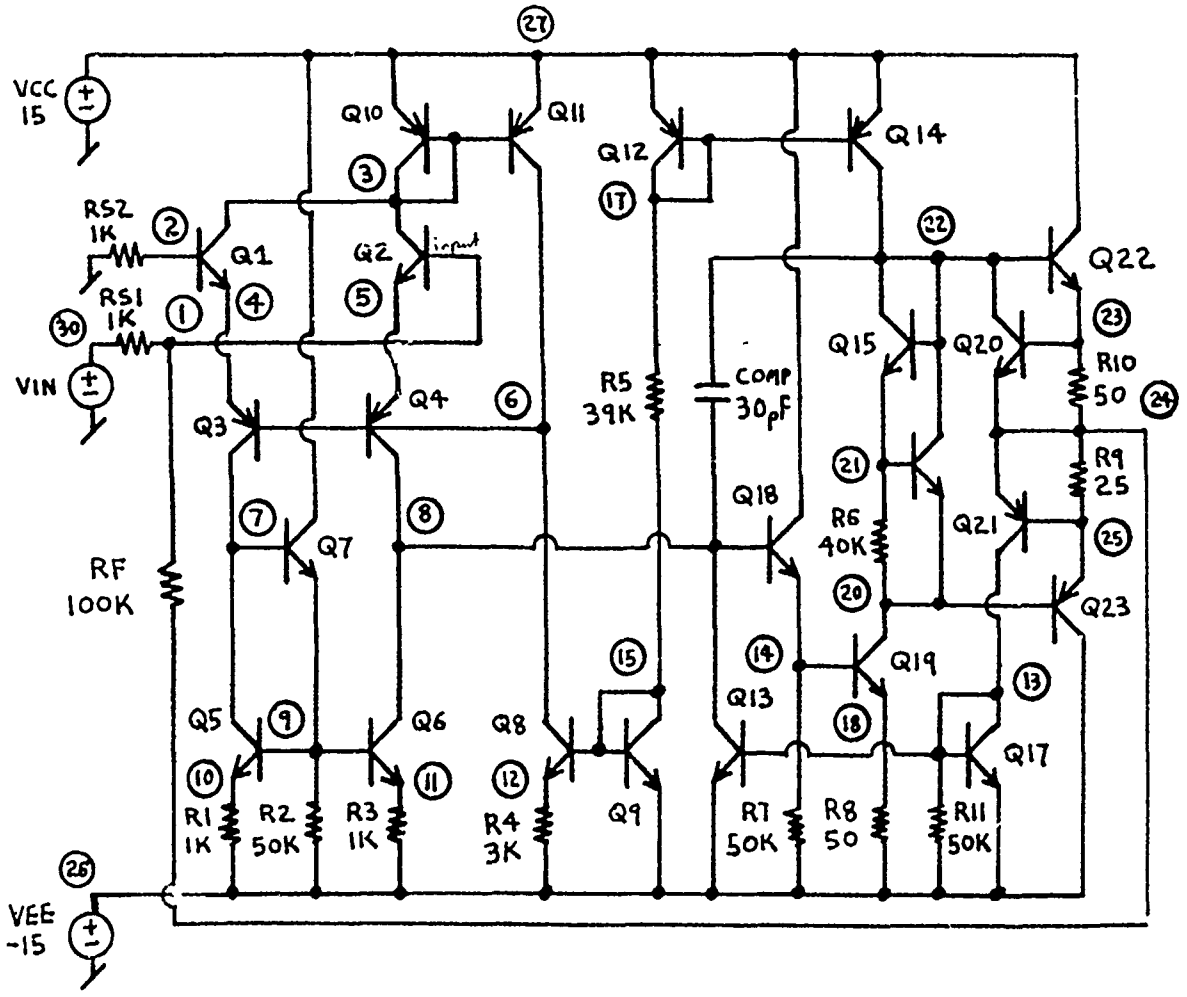



***Figure 2-5C.*** UA741 Op-amp circuit (from ref. 4).

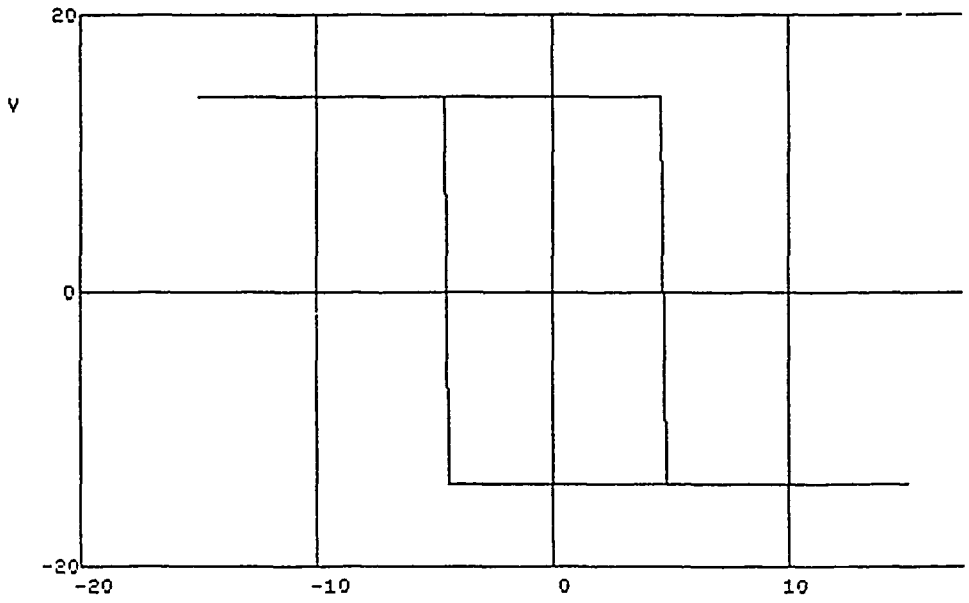



***Figure 2-5D.*** Response of comparator circuit when input from -15 V to +15V and back to -15 V is applied as shown in Figure 2-5A.

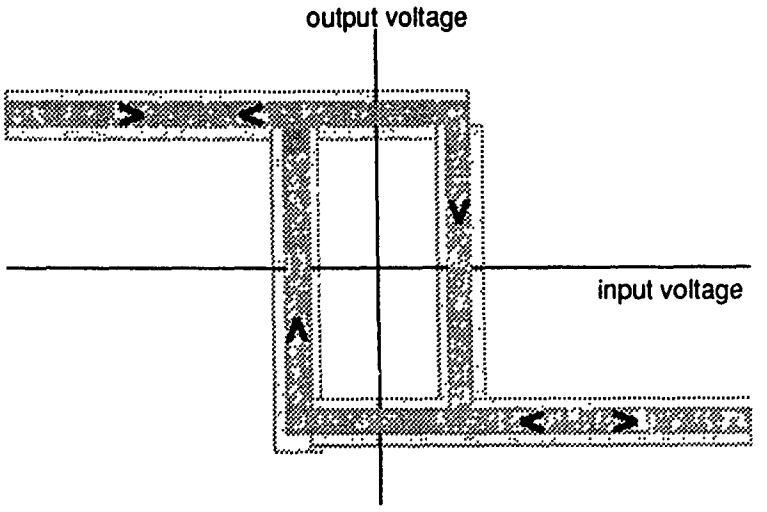



***Figure 2-5E.*** Probabilistic response of the comparator when components have nominal variation.

## 2.5B. Analysis

The circuit of Figure 2-5A was sent to the SFA. Two-hundred "nominal" Monte Carlo runs were performed, along with 100 Monte Carlo runs representing each of the fault cases arising from the "opening" and "shorting" of resistors $R_1$, $R_f$, and $R_{in}$. Data collected were the output voltages from the comparator (node 6) resulting from the input voltage ramping. For each Monte Carlo run, a full input ramp was performed (resulting in 30 output voltages per run). This should result in a "cloudy" hysteresis loop when the results of all nominal runs are plotted (Figure 2-5E). When faults occur, the switching points in the loops presumably should shift.

Using the SAS STEPDISC procedure, voltages that do not contribute to the discrimination power of the discrimination analysis (following Wilk's Lambda tests of Section 1.5C) were eliminated. Of the original thirty input voltages for each run, all but 12 voltages were eliminated. Those that were kept reflected the shifting of the switch points in the hysteresis curve.

The results from 100 test runs for each of the seven fault class were then applied in the discrimi-

nation and hypothesis tests. The discrimination analysis yielded the following results:

| | |
|---|---|
| Nominal | 100% correct classification |
| $R_1$ open | 3% misclassified as nominal |
| $R_f$ open | 2% misclassified as nominal |
| $R_{in}$ open | 12% misclassified as nominal |
| others | 100% correctly classified |

The hypothesis test (eq. 1-12) results for the comparator are as follows:

| | |
|---|---|
| Nominal | 94% classified as "good" |
| Faults | 100% correctly rejected as "bad" |

In summary, almost all of the nominal circuits were classified as "good." This means that assuming production runs of the comparator were identical to the simulation results, few "good" circuits would be mistakenly rejected as being "bad." However, a few per cent of "bad" circuits would be erroneously passed as "good" following the discrimination analysis. The severe incorrect acceptance of the "Rin open" fault cases is to be expected, given the high input impedance of the op-amp. If the voltage drop across this device were measured, this classification error would likely be eliminated.

## 2.6. Logarithmic Amplifier

### 2.6A. Circuit Description

Figure 2-6A describes the circuit of this part of our study. The configuration is that of a *log ratio* operator [3]. When two voltages, $v_1$ and $v_2$, are applied to the circuit, the output is given by the transfer function:

$$v_0 = -K_1 ln\left(\frac{K_2 v_1}{v_2}\right)$$

with

$$K_1 = \frac{R_1 + R_2}{R_2} \cdot \frac{kT}{q}$$

and

$$K_2 = \frac{R_3}{R_5}$$

For the usual log operator function, $v_2$ is typically held constant at some reference voltage. In our simulations, we fixed this value at +15 V (which could be conveniently drawn from the supply). An important advantage of this circuit over the simpler one op-amp / one-transistor log amplifier circuit is that it is more tolerant to temperature drift of the components. Further temperature compensation is provided by resistor $R_2$, this component being temperature compensated. Figure 2-6B shows the result of a SPICE simulation, where the input $v_1$ was ramped from 0.01 to 10 V. The presence of the exponential function in the dependent voltage source of the SFA/SPICE deck (see Figure 2-6C) enables SPICE's DC routine to ramp with an exponentially increasing input voltage over three decades. Note that highly linear logarithmic response over the 3-decade range of the simulations (which used the linearized 741 op-amp model).

### 2.6B Analysis

Two hundred nominal Monte Carlo runs were performed, followed by 10 runs for each fault class, which were defined by opening and shorting all of the transistor resistances and component resistors in the circuit. All runs consisted of ramping the input voltage exponentially from 0.01 to 10 V. Voltages at nodes 2, 6, and 8 were recorded (resulting in a total of 90 points per run). Each

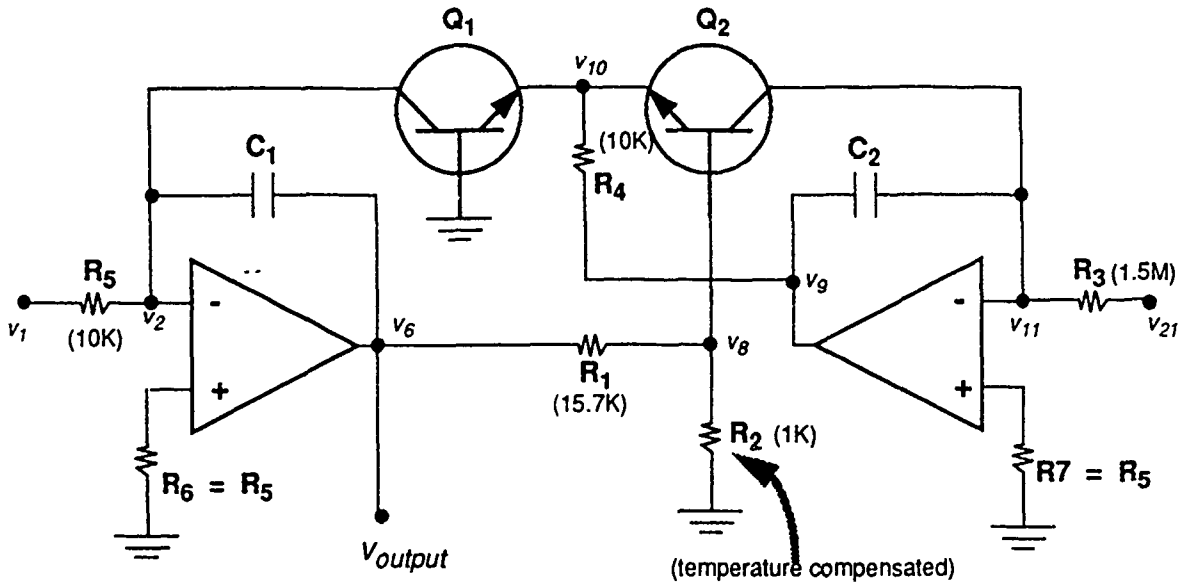



*Figure 2-6A.* Log-ratio operator circuit.

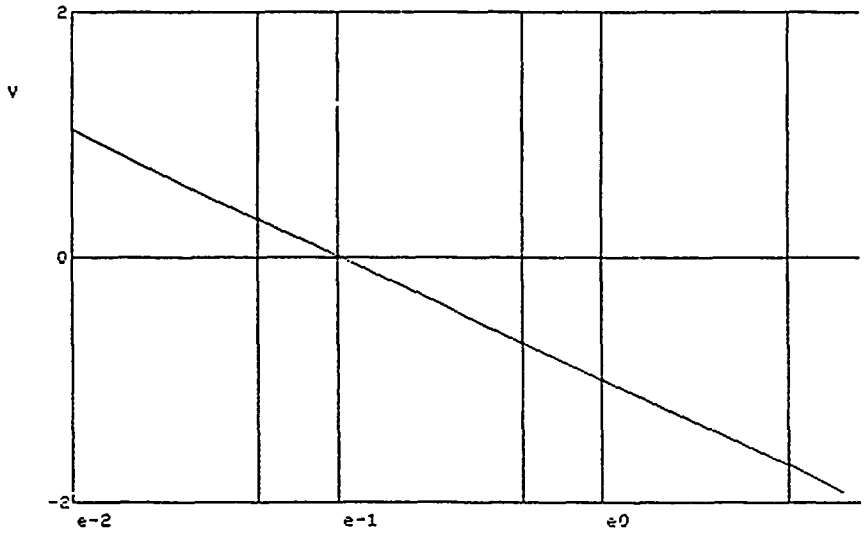



*Figure 2-6B.* SPICE response of the log-ratio amplifier. An input voltage ranging from 0.01 to 10 V was applied to $v_1$.

```
Log-ratio circuit
* B. Epstein  7/90
#GOOD 200
#RB1 1.0 10 RB1_SH
#RC1 10000.0 10 RC1_OP
#RE1 10000.0 10 RE1_OP
#RBE1 0.01 10 RBE1_SH
#RCE1 0.01 10 RCE1_SH
#RBC1 0.0001 10 RBC1_SH
#QNL1 1.0 10 Q1_LOW BF
#RB2 1.0 10 RB2_SH
#RC2 10000.0 10 RC2_OP
#RE2 10000.0 10 RE2_OP
#RBE2 0.01 10 RBE2_SH
#RCE2 0.01 10 RCE2_SH
#RBC2 0.0001 10 RBC2_SH
#QNL2 1.0 10 Q2_LOW BF
#RM5 100000.0 10 R5_OP
#RM6 100000.0 10 R6_OP
#RM2 0.01 10 R2_SH
#RM2 100000.0 10 R2_OP
#RM7 0.0001 10 R7_SH
#RM3 0.0001 10 R3_SH
#RM1 0.001 10 R1_SH
#RM1 100000.0 10 R1_OP
#RM4 0.001 10 R4_SH
#RM4 100000.0 10 R4_OP
%dc 2 6 8
.SUBCKT UA741 2  3  6   7   4
*             - IN + OUT VCC VEE
RP 4 7 [10.0,0.5]K
RXX 4 0 [10.0,0.5]MEG
IBP 3 0 [80.0,4.0]NA
RIP 3 0 [10.0,0.5]MEG
CIP 3 0 [1.4,0.07]PF
IBN 2 0 [100.0,5.0]NA
RIN 2 0 [10.0,0.5]MEG
CIN 2 0 [1.4,0.07]PF
VOFST 2 10 [1.0,0.05]MV
RID 10 3 [200.0,10.0]K
EA 11 0 10 3 1
R1 11 12 [5.0,.25]K
R2 12 13 [50,2.5]K
C1 12 0 [13.0,.65]PF
GA 0 14 0 13 [2700.0,135]
C2 13 14 [2.7,.14]PF
RO 14 0 [75.0,3.75]
L 14 6 [30.0,1.5]UHY
RL 14 6 [1000,50.0]
CL 6 0 [3.0,.15]PF
.ENDS UA741
*
.SUBCKT QLOG1 1 2 3
Q1 11 12 13 QNL1
RC1 1 11 [0.1,0.0]
RB1 2 12 [100.0,0.0]
RE1 3 13 [0.1,0.0]
RBE1 12 13 [10000.0,0.0]K
RCE1 11 13 [10000.0,0.0]K
RBC1 11 12 [1000.0,0.0]MEG
.MODEL QNL1 NPN (BF=[80.0,8.0]
+ CCS=2PF TF=0.2NS TR=6NS CJE=3PF
+  CJC=2PF VA=[50.0,5.0])
ENDS QLOG1
*
.SUBCKT QLOG2 1 2 3
Q2 11 12 13 QNL2
RC2 1 11 [0.1,0.0]
RB2 2 12 [100.0,0.0]
RE2 3 13 [0.1,0.0]
RBE2 12 13 [10000.0,0.0]K
RCE2 11 13 [10000.0,0.0]K
RBC2 11 12 [1000.0,0.0]MEG
.MODEL QNL2 NPN (BF=[80.0,8.0]
+ CCS=2PF TF=0.2NS TR=6NS CJE=3PF
+  CJC=2PF VA=[50.0,5.0])
.ENDS QLOG2
*
* MAIN CIRCUIT
VCC 7 0 15.0
VEE 4 0 -15.0
V1IN 25 0 DC
B1 20 0 V=exp(V(25)/0.433)
V2IN 21 0 15.0
RM5 20 2 [10.0,0.5]K
RM6 3 0 [10.0,0.5]K
XQ1 2 0 10 QLOG1
XQ2 11 8 10 QLOG2
RM2 8 0 [1.0,0.05]K
RM7 13 0 [1.5,0.07]MEG
RM3 21 11 [1.5,0.07]MEG
RM1 6 8 [15.7,0.7]K
RM4 10 9 [10.0,0.5]K
X1 2 3 6 7 4 UA741
X2 11 13 9 7 4 UA741
.dc V1IN -2 1 .1
.END
```

***Figure 2-6C.*** Log-ratio circuit (based on ref. 3)

of the 10 fault runs were used to train the discriminator. Using the SAS DISCRIM routine, the following results were obtained, where combinations of selected nodes and input voltages appear in Table 2-6A:

Discrimination Results

Using only 10 voltages (see Table 2-6A, Set A):

100% of nominal correctly classified as nominal
100% of the following faults incorrectly classified as nominal:
"R4 short" "R7 short" "RB2 short" "RC2 open"
100% of all other fault classes properly classified

Using node 2 @ Vin of 0.0099 V and node 6 @ 0.79 V:

only 1.5% of nominals correctly classified

Using nodes 2, 6, and 8 @ Vin of 0.0099 V and 1.00 V:

28% of nominals correctly classified; remaining nominals confused with:
"R4 short" "R7 short" "Rb1 short" "RB2 short" "RBC2 open"

Using SAS STEPDISC reduction to 10 voltages (Table 2-6A, Set B):

82.5% of nominal correctly classified as nominal; remaining confused with:
"R7 short" "RB2 short" "RC2 short"
Most fault classes correctly classified except "R4 short" and "R7 short"

Following the hypothesis testing procedure of eq. 1-12, the following results occurred:

Hypothesis Testing

Using the reduced STEPDISC set (Table 2-6A, Set B):

186 out of 200 (or 93%) of nominals accepted
190 of 240 faults correctly rejected
Faults incorrectly accepted:
"RE1 open" "RC1 open" "R3 short" "R4 open" "Q1 low"

Given the limited number of nodes for this analysis (only nodes 2, 6, and 8), we have arrived at an unsatisfactory number of false acceptances of what are otherwise "bad" circuits. This applies to both the discrimination analysis and hypothesis test results. Additional nodes, such as nodes 9, 10, and 11 may have to be included. In contrast, no "good" circuits were falsely rejected.

**Table 2-6A. Analysis Voltage Sets (for log-ratio circuit)**

| Set A | | Set B | |
|---|---|---|---|
| *node* | *Vin (V)* | *node* | *Vin (V)* |
| 2 | 0.0093 | 6 | 0.00986 |
| 2 | 0.0156 | 8 | 0.0394 |
| 8 | 0.0394 | 8 | 0.0497 |
| 6 | 0.0788 | 8 | 0.0626 |
| 2 | 0.198 | 2 | 0.158 |
| 8 | 0.397 | 2 | 0.199 |
| 6 | 0.794 | 8 | 0.794 |
| 2 | 2.00 | 6 | 0.794 |
| 8 | 5.04 | 2 | 1.00 |
| 6 | 7.99 | 8 | 1.00 |

## 2.7. Analog Multiplexer

### 2.7A. Circuit Description

This circuit represents a "mixed-technology" design, where MOS transistors serve as switches that route signals to a bipolar 741 op-amp that is wired in a conventional noninverting summation configuration (Figure 2-7A). Eight inputs drive the multiplexer. We assume that proper operation calls for at most one channel to be "on" at a given time. The SFA input file is given in Figure 2-7B. The circuit was simulated using the full 23 transistor 741 op-amp SPICE description (Figure 2-6C). Each MOS switch was turned on by applying +15 V to a given device's gate; all remaining MOS gates were held at -15 V to keep them in an off state. The control voltages of +15 and -15 V were chosen for convenience since these are also the op-amp's supply voltages. A test voltage of +5 V was applied to each of the multiplexer's inputs; i.e., to the drains of the MOS switches.

### 2.7B. Analysis

Unlike all of our previous circuit analyses, the multiplexer presents a situation where there exists a multitude of "good" circuit operating states. Each of the "good" states corresponds to at most one gate "on" while all other gates are "off". In the case of an 8-input analog mux, we then have 9 nominal states to consider -- the "all gates off" state plus each of the eight "one-gate-on" states. Upon introducing a fault to the circuit, we must then attempt to see if the circuit's behavior resembles any of the "good" states. If not, the circuit is rejected as "bad" in a Go/No-Go test. If a discrimination analysis is performed, an attempt to identify the fault is also made.

In our analysis of the mux circuit, 100 Monte Carlo runs were performed for each of the "on" states and the "all-gates-off" state. We then "turned on" channel 3 (by applying +15 V to its gate) and introduced various faults to the circuit while input remained on. Channel 3 was arbitrarily chosen. Initially, we attempted to use all nodal voltages indicated in Figure 2-7A(refer also to the *%dc* line in 2-7B). However, the SAS STEPDISC procedure reduced the useful measurement nodes to only nodes 3 and 6.

Comparing the faulted circuit responses (each fault having 10 runs) with the "3 ON" state, the SAS DISCRIM routine gave the following results:

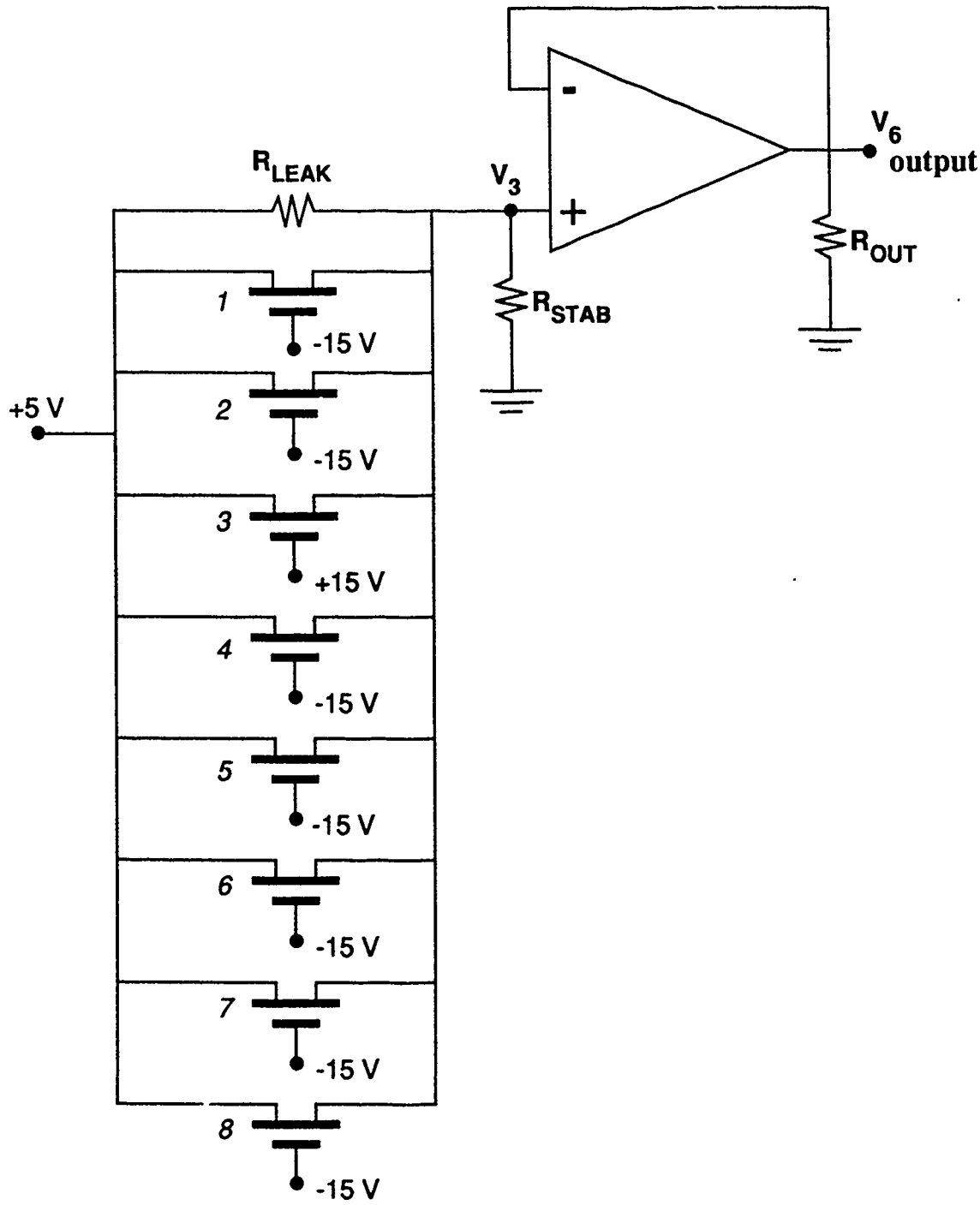



***Figure 2-7A.*** Analog multiplexer using a noninverting summation op-amp configuration.

```
Analog multiplexer
* B. Epstein  7/90
#V1 -15.0 100 ALL_OFF
#V1 15.0 100 1_ON
#V2 15.0 100 2_ON
#V3 15.0 100 3_ON
#V4 15.0 100 4_ON
#V5 15.0 100 5_ON
#V6 15.0 100 6_ON
#V7 15.0 100 7_ON
#V8 15.0 100 8_ON
%dc 6 3 21 22 23 24 25 26 27 28
*
.SUBCKT UA741 1   2  24   27 26
*             (-) (+) OUT  V+ V-
R1 10 26 [1.0,.05]K
R2 9 26 [50.0,2.5]K
R3 11 26 [1.0,0.05]K
R4 12 26 [3.0,0.15]K
R5 15 17 [39.0,2.0]K
R6 21 20 [40.0,2.0]K
R7 14 26 [50.0,2.5]K
R8 18 26 [50.0,2.5]
R9 24 25 [25.0,1.25]
R10 23 24 [50.0,2.0]
R11 13 26 [50.0,2.0]K
COMP 22 8 30PF
Q1 3 2 4 QNL
Q2 3 1 5 QNL
Q3 7 6 4 QPL
Q4 8 6 5 QPL
Q5 7 9 10 QNL
Q6 8 9 11 QNL
Q7 27 7 9 QNL
Q8 6 15 12 QNL
Q9 15 15 26 QNL
Q10 3 3 27 QPL
Q11 6 3 27 QPL
Q12 17 17 27 QPL
Q13 8 13 26 QNL
Q14 22 17 27 QPL
Q15 22 22 21 QNL
Q16 22 21 20 QNL
Q17 13 13 26 QNL
Q18 27 8 14 QNL
Q19 20 14 18 QNL
Q20 22 23 24 QNL
Q21 13 25 24 QPL
Q22 27 22 23 QNL
Q23 26 20 25 QPL
.MODEL QNL NPN(BF=[80.0,8.0]
+ RB=[100.0,5.0] CCS=2PF TF=0.2NS
+ TR=6NS CJE=3PF
+  CJC=2PF VA=[50.0,5.0])
.MODEL QPL PNP(BF=[10.0,1.0]
+ RB=[20.0,1.0] TF=1NS TR=20NS
+ CJE=6PF CJC=4PF VA=[50.0,5.0])
.ENDS
*
* MAIN CIRCUIT
VCC 7 0 15.0
VEE 4 0 -15.0
Vset 1 0 5.0
* Control voltages:
V1 21 0 [-15.0,0.0]
V2 22 0 [-15.0,0.0]
V3 23 0 [-15.0,0.0]
V4 24 0 [-15.0,0.0]
V5 25 0 [-15.0,0.0]
V6 26 0 [-15.0,0.0]
V7 27 0 [-15.0,0.0]
V8 28 0 [-15.0,0.0]
M1 1 21 3 4 MOD1 L=[4.0,0.04]U W=[6.0,0.06]
+ U AD=[10.0,0.1]P AS=[10.0,0.1]P
M2 1 22 3 4 MOD1 L=[4.0,0.04]U W=[6.0,0.06]
+ U AD=[10.0,0.1]P AS=[10.0,0.1]P
M3 1 23 3 4 MOD1 L=[4.0,0.04]U W=[6.0,0.06]
+ U AD=[10.0,0.1]P AS=[10.0,0.1]P
M4 1 24 3 4 MOD1 L=[4.0,0.04]U W=[6.0,0.06]
+ U AD=[10.0,0.1]P AS=[10.0,0.1]P
M5 1 25 3 4 MOD1 L=[4.0,0.04]U W=[6.0,0.06]
+ U AD=[10.0,0.1]P AS=[10.0,0.1]P
M6 1 26 3 4 MOD1 L=[4.0,0.04]U W=[6.0,0.06]
+ U AD=[10.0,0.1]P AS=[10.0,0.1]P
M7 1 27 3 4 MOD1 L=[4.0,0.04]U W=[6.0,0.06]
+ U AD=[10.0,0.1]P AS=[10.0,0.1]P
M8 1 28 3 4 MOD1 L=[4.0,0.04]U W=[6.0,0.06]
+ U AD=[10.0,0.1]P AS=[10.0,0.1]P
.MODEL MOD1 NMOS VTO=-[2.0,0.1]
+  NSUB=[1.0,0.05]E15 UO=[550.0,27.5]
X1 6 3 6 7 4 UA741
Rout 6 0 10K
Rstab 3 0 10K
.op
* .dc Vset -5.0 5.0 1.0
.END
```

***Figure 2-7B.*** Analog multiplexer circuit. This file establishes nominal operation.

```
Analog multiplexer
* B. Epstein 7/90
#Rleak 0.001 10 R_LK1
#Rleak 0.1 10 R_LK100
#Rleak 1.0 10 R_LK1K
#Rleak 5.0 10 R_LK5K
#Rleak 10.0 10 R_LK10K
#Rout 0.01 10 RO_10
#Rstab 0.1 10 RST_100
#QNL 200.0 10 QNL_hi BF
#QNL 5.0 10 QNL_lo BF
#QPL 10000.0 10 QPL_BOPEN RB
#MOD1 0.01 10 MOD1_LOW VTO
#MOD1 10.0 10 UO_LOW UO
#R3 0.001 10 R3_short
#R10 10000.0 10 R10_hi
%dc 6 3 21 22 23 24 25 26 27 28
*
.SUBCKT UA741 1 2 24 27 26
* (-) (+) OUT V+ V-
R1 10 26 [1.0,.05]K
R2 9 26 [50.0,2.5]K
R3 11 26 [1.0,0.05]K
R4 12 26 [3.0,0.15]K
R5 15 17 [39.0,2.0]K
R6 21 20 [40.0,2.0]K
R7 14 26 [50.0,2.5]K
R8 18 26 [50.0,2.5]
R9 24 25 [25.0,1.25]
R10 23 24 [50.0,2.0]
R11 13 26 [50.0,2.0]K
COMP 22 8 30PF
Q1 3 2 4 QNL
Q2 3 1 5 QNL
Q3 7 6 4 QPL
Q4 8 6 5 QPL
Q5 7 9 10 QNL
Q6 8 9 11 QNL
Q7 27 7 9 QNL
Q8 6 15 12 QNL
Q9 15 15 26 QNL
Q10 3 3 27 QPL
Q11 6 3 27 QPL
Q12 17 17 27 QPL
Q13 8 13 26 QNL
Q14 22 17 27 QPL
Q15 22 22 21 QNL
Q16 22 21 20 QNL
Q17 13 13 26 QNL
Q18 27 8 14 QNL
Q19 20 14 18 QNL
Q20 22 23 24 QNL
Q21 13 25 24 QPL
Q22 27 22 23 QNL
Q23 26 20 25 QPL
.MODEL QNL NPN(BF=[80.0,8.0]
+ RB=[100.0,5.0] CCS=2PF TF=0.2NS
+ TR=6NS CJE=3PF
+ CJC=2PF VA=[50.0,5.0])
 MODEL QPL PNP(BF=[10.0,1.0]
+ RB=[20.0,1.0] TF=1NS TR=20NS
+ CJE=6PF CJC=4PF VA=[50.0,5.0])
.ENDS
*
* MAIN CIRCUIT
VCC 7 0 15.0
VEE 4 0 -15.0
Vset 1 0 5.0
* Control voltages:
V1 21 0 -15.0
V2 22 0 -15.0
V3 23 0 15.0
V4 24 0 -15.0
V5 25 0 -15.0
V6 26 0 -15.0
V7 27 0 -15.0
V8 28 0 -15.0
Rleak 1 3 [100000.0,0.0]K
M1 1 21 3 4 MOD1 L=[4.0,0.04]U
+ W=[6.0,0.06]U AD=[10.0,0.1]P
+ AS=[10.0,0.1]P
M2 1 22 3 4 MOD1 L=[4.0,0.04]U
+ W=[6.0,0.06]U AD=[10.0,0.1]P
+ AS=[10.0,0.1]P
M3 1 23 3 4 MOD1 L=[4.0,0.04]U
+ W=[6.0,0.06]U AD=[10.0,0.1]P
+ AS=[10.0,0.1]P
M4 1 24 3 4 MOD1 L=[4.0,0.04]U
+ W=[6.0,0.06]U AD=[10.0,0.1]P
+ AS=[10.0,0.1]P
M5 1 25 3 4 MOD1 L=[4.0,0.04]U
+ W=[6.0,0.06]U AD=[10.0,0.1]P
+ AS=[10.0,0.1]P
M6 1 26 3 4 MOD1 L=[4.0,0.04]U
+ W=[6.0,0.06]U AD=[10.0,0.1]P
+ AS=[10.0,0.1]P
M7 1 27 3 4 MOD1 L=[4.0,0.04]U
+ W=[6.0,0.06]U AD=[10.0,0.1]P
+ AS=[10.0,0.1]P
M8 1 28 3 4 MOD1 L=[4.0,0.04]U
+ W=[6.0,0.06]U AD=[10.0,0.1]P
+ AS=[10.0,0.1]P
.MODEL MOD1 NMOS VTO=-[2.0,0.1]
+ NSUB=[1.0,0.05]E15 UO=[550.0,27.5]
X1 6 3 6 7 4 UA741
Rout 6 0 [10.0,0.0]K
Rstab 3 0 [10.0,0.0]K
.op
* .dc Vset -5.0 5.0 1.0
.END
```

***Figure 2-7B (cont.).*** Analog multiplexer circuit. This file establishes faulted operation with channel 3 "on."

Discrimination Analysis (14 fault cases; 10 runs per fault)

100% of following faults were correctly classified (refer to second file in Figure 2-7B for identification):

MOD1 low
QNL low
R3 shorted
RO 10
RST 100
RLK 1
RLK 100
RLK 1K
RLK 5K
RLK 10K

The remaining faults were confused with the nominal "3 ON" state (percentages following fault are rate of correct classification):

QNL high(50%)
QPL open base(70%)
R10 high(20%)
UO low(50%)

Only 24% of the 100 "3 ON" runs were correctly classified as nominal, the remainder being classified among the above four faults.

SAS's nonparametric NEIGHBOR routine, which classifies the faults according to a nearest-neighbor rule, yielded slightly better results. The following faults were confused with the nominal "3 ON" state:

Nearest Neighbor Discrimination (14 fault cases; 10 runs per fault)

QNL high(20%)
QPL open base(80%)
R10 high(20%)
UO low(10%)

78% of "3 ON" nominal cases were correctly classified as nominal.

The nearest neighbor method had a better success of classifying the nominal "3 ON" runs, but the fault classification was slightly worse overall compared to the conventional probabilistic discrimination.

The hypothesis testing for Go/No-Go testing (by eq. 1-12) resulted in the following:

Hypothesis Testing (14 fault cases; 10 runs per fault)

100% of the circuits with the following faults led to incorrect acceptance:
R10 high
UO low
QPL open base
QNL high

All remaining faulted circuits were correctly rejected as "bad"
96 of 100 "good" "3-ON" circuits were correctly accepted

Of the 14 fault cases introduced, four types of faults could not be separated from the "3-ON" nominal case. This ambiguity occurred in both the hypothesis testing and the two forms of discriminant analysis. Given that only two nodal voltages were used, this result is quite impressive but not adequate for any production test. More test nodes are likely required, particularly within the op-amp.

### 2.7C. Analog Multiplexer Summary

Recall from our previous circuits that the use of the discrimination analysis for performing Go/No-Go testing is a risky procedure. This is because there may exist faults that were not accounted for during the construction of the discrimination algorithm. The introduction of unknown faults not accounted for during the construction of the discriminator can result in an ambiguous classification of the unknown fault -- in particular classification as a "nominal" circuit. In contrast, the simple hypothesis test was constructed only with nominal runs. Therefore, unknown faults can be introduced into the test. Of course, this test provides no information on the specific cause of the fault.

In the case of a multistate circuit, such as the analog multiplexer, the response of a given circuit under test could be subjected to a series of hypothesis tests where each test corresponds to one of the many possible "good" states. For example, a test sequence for the 8 channel mux may call for turning on each of the eight channels one at a time, and applying a hypothesis test based on the assumed nominal response of the circuit with the same channel turned on. In the analysis of this report, we only checked for the proper operating condition of the mux circuit with channel 3 on. A full test would require that this test be repeated for each of the remaining seven channels.

## 2.8. D/A Converter

### 2.8A. Circuit Description

The digital-to-analog (D/A) converter of this study makes use of a bank of current sources whose currents are summed and converted to a voltage by an op-amp circuit. Each current source is set on or off according to the bit pattern submitted to the circuit. Figure 2-8A shows the overall circuit, which provides 4 bits of resolution. The circuit's current sources were derived from a high-speed video D/A design developed at the David Sarnoff Research Center. One current source, supplying current $I_0$, is shown in Figure 2-8B. The circuit makes use of PMOS devices. Transistors $P_1$ and $P_2$ serve as current sources, while transistors $P_3$ and $P_4$ "steer" the current either to ground or the output port $I_{OUT}$. Current is routed to Iout when +5V is applied to terminal $V_c$. When zero volts is applied to $V_c$, the current through $I_{OUT}$ is approximately zero. The magnitude of the current through $I_{OUT}$ is set by the voltage difference between $V_{DD}$ and $V_R$. Note that when the circuit is in the "off" state (i.e., $I_{OUT} = 0$), the effective impedance at the $I_{OUT}$ terminal is high to prevent the loading of other current sources in the D/A converter.

The circuit of Figure 2-8B is set to provice $I_0 = 44.2\ \mu A$ to a 500 Ω load. By eliminating $P_2$, 1/2 $I_0$ is achieved. Connecting $P_1$ and $P_2$ in series results in 1/4 $I_0$ (refer to SFA/SPICE file in Figure 2-8C for the configuration of these modifications to the current sources). The SPICE MOS model sheet resistance parameter RSH was set to the default value (10 Ω/sq.) since other values caused the program to crash.

The op-amp is wired in a current-summation / inverting amplifier configuration. The amplifier's gain was set (by $R_f$) to produce an output of roughly 0.1V per bit, or -1.5V full scale (for bit pattern 1111). For convenience during the simulations, the op-amp was modeled using the full 741 model. In practice, however, the speed and offset characteristics of this op-amp do not match the superior switching characteristics of the current sources, thus a high performance D/A converter would not use this op-amp.

## 2.8B. Fault Analysis

Faults were introduced into the circuit by changing the setting of the variable resistors $R_{short}$ and $R_{open}$ (Figure 2-8A). The shorting of each current source to ground was represented by setting a current source's respective $R_{short}$ to 1 Ω. Likewise, the setting of a current source's corresponding $R_{open}$ to 10,000 MΩ effectively opened that current source.

To create as broad a fault coverage as possible without having to resort toa combinatorial explosion of test steps, we introduced the simple test pattern of sequentially turning on each bit from the least significant to the most significant; i.e., 0000, 0001, 0011, 0111, 1111. For each bit pattern, each open and short condition was then applied to the circuit. Only one fault at a time was considered. Fifty Monte Carlo runs were applied for each fault condition. "Nominal" operation was defined as circuit operation without any faults applied. The overall procedure was applied with a one sigma variation of 5% and 2.5%.

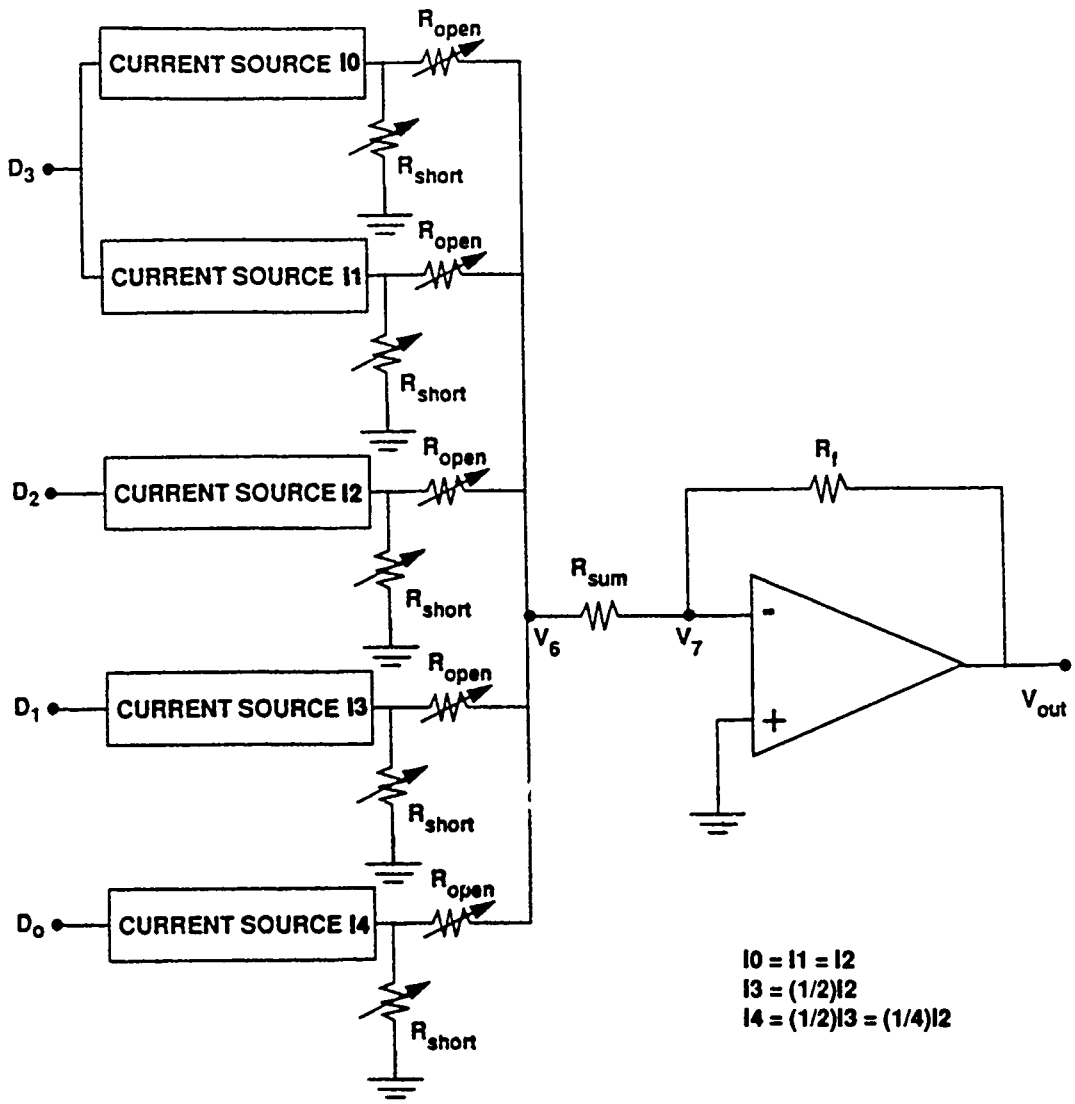



*Figure 2-8A.* 4-bit D/A converter. The current sources are switched on by application of 5V to the control inputs Dn. The variable resistors represent the application of shorts and opens to the current source outputs.

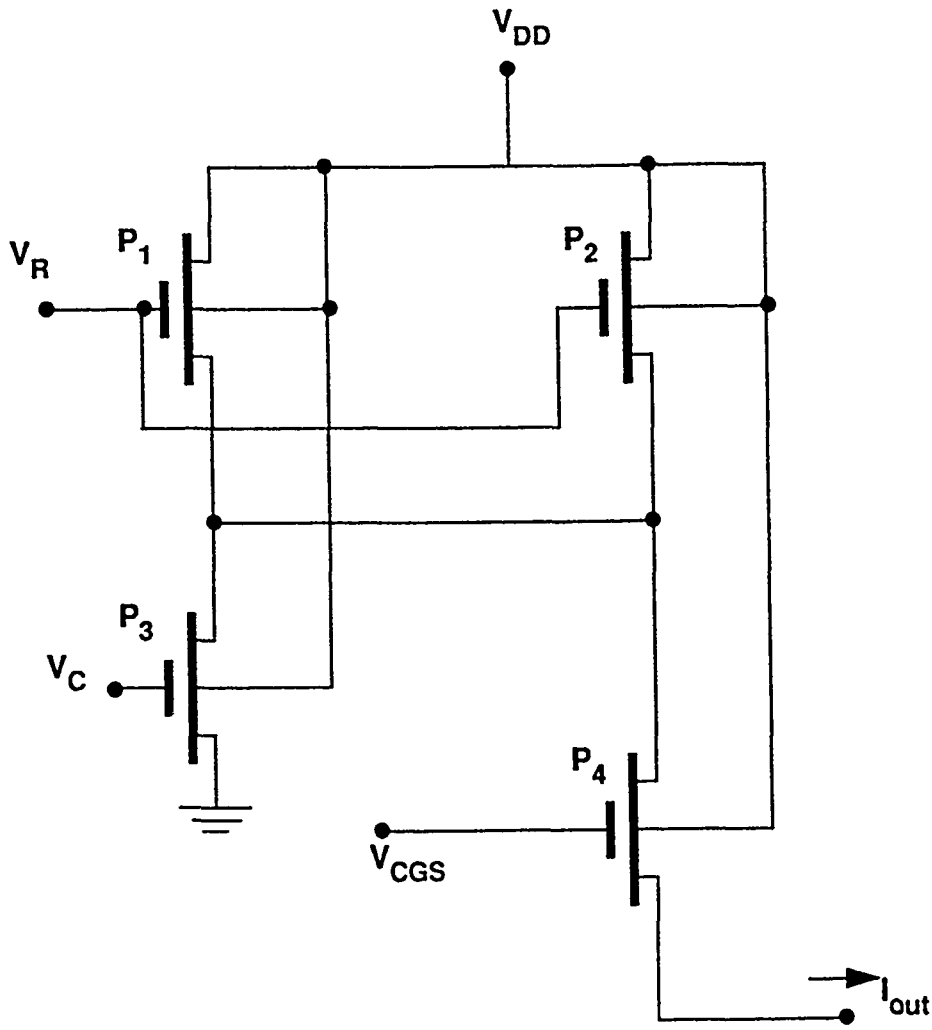



***Figure 2-8B.*** Current source used in D/A converter of Figure 2-8A. PMOS devices are used. See text for details of operation (*source: David Sarnoff Research Center*).

```
PMOS/Bipolar D-A Converter
* Uses Sarnoff current sources
* B. Epstein 8/90
#GOOD 50
#RXO0 10000.0MEG 50 I0P0111
#RXO1 10000.0MEG 50 I1P0111
#RXO2 10000.0MEG 50 I2P0111
#RXO3 10000.0MEG 50 I3P0111
#RXO4 10000.0MEG 50 I4P0111
#RXS0 .001 50 IOS0111
#RXS1 .001 50 I1S0111
#RXS2 .001 50 I2S0111
#RXS3 .001 50 I3S0111
#RXS4 .001 50 I4S0111
%dc 7 8 20 21 22 23
* Full current source:
.SUBCKT ISOURCE 1 2 4 5 6
* VDD1 VR VC VGCS Iout
MP1 3 2 1 1 PE L=[4.0,0.2]U W=[8.0,0.4]U
MP2 3 2 1 1 PE L=[4.0,0.2]U W=[8.0,0.4]U
MP3 0 4 3 1 PE L=[1.3,0.07]U W=[8.0,0.4]U
MP4 6 5 3 1 PE L=[2.0,0.1]U W=[8.0,0.4]U
.ENDS ISOURCE
*
* 1/2 current PMOS source:
.SUBCKT ISOURCE2 1 2 4 5 6
* VDD1 VR VC VGCS Iout
MP1 3 2 1 1 PE L=[4.0,0.2]U W=[8.0,0.4]U
MP3 0 4 3 1 PE L=[1.3,0.07]U W=[6.0,0.3]U
MP4 6 5 3 1 PE L=[2.0,0.1]U W=[6.0,0.3]U
.ENDS ISOURCE2
*
* 1/4 current PMOS source:
.SUBCKT ISOURCE4 1 2 4 5 6
* VDD1 VR VC VGCS Iout
MP1 3 2 1 1 PE L=[4.0,0.2]U W=[8.0,0.4]U
MP2 8 2 3 1 PE L=[4.0,0.2]U W=[8.0,0.4]U
MP3 0 4 8 1 PE L=[1.3,0.07]U W=[4.0,0.2]U
MP4 6 5 8 1 PE L=[2.0,0.1]U W=[6.0,0.3]U
.ENDS ISOURCE4
*
* PMOS Switch Model:
.MODEL      PE      PMOS(LEVEL=3      VTO=[-0.75,0.07]
TOX=[250.0,12.0]E-10
+ NSUB=1.3E16 LD=0.0 UO=190.0
+ VMAX=8.366E5 THETA=0.175 ETA=0.20 KAPPA=3.335
+ XJ=0.45E-6 NFS=1.0E12 TPG=1 DELTA=1.2
+ CGSO=2.5E-10 CGDO=2.5E-10 CJ=3.18E-4 CJSW=1.97E-10
+ MJ=0.57 MJSW=0.12 PB=0.8)
*
* Op-Amp (linearized 741 model)
.SUBCKT UA741 2 3 6 7 4
* - IN + OUT VCC VEE
RP 4 7 [10.0,0.5]K
RXX 4 0 [10.0,0.5]MEG
IBP 3 0 [80.0,4.0]NA
RIP 3 0 [10.0,0.5]MEG
CIP 3 0 [1.4,0.07]PF
IBN 2 0 [80.0,5.0]NA
RIN 2 0 [10.0,0.5]MEG
CIN 2 0 [1.4,0.07]PF
VOFST 2 10 [1.0,0.05]MV
RID 10 3 [200.0,10.0]K
EA 11 0 10 3 1
R1 11 12 [5.0,.25]K
R2 12 13 [50,2.5]K
C1 12 0 [13.0,.65]PF
GA 0 14 0 13 [2700.0,135]
C2 13 14 [2.7,.14]PF
RO 14 0 [75.0,3.75]
L 14 6 [30.0,1.5]UHY
RL 14 6 [1000,50.0]
CL 6 0 [3.0,.15]PF
.ENDS UA741
*
* *** MAIN CIRCUIT ***
*
* Voltages:
VCC 10 0 15.0
VEE 11 0 -15.0
VDD1 1 0 [5.0,0.05]V
VR 2 0 [3.2,0.01]V
VGCS 5 0 [1.7,0.02]V
VC0 20 0 0.0
VC1 21 0 5.0
VC2 22 0 5.0
VC3 23 0 5.0
*
* Connections:
X0A 1 2 20 5 30 ISOURCE
X0B 1 2 20 5 31 ISOURCE
X1 1 2 21 5 32 ISOURCE
X2 1 2 22 5 33 ISOURCE2
X3 1 2 23 5 34 ISOURCE4
* Catastrophic Faults:
RXO0 30 6 [0.5,0.0]
RXO1 31 6 [0.5,0.0]
RXO2 32 6 [0.5,0.0]
RXO3 33 6 [0.5,0.0]
RXO4 34 6 [0.5,0.0]
RXS0 30 0 [10000.0,0.0]K
RXS1 31 0 [10000.0,0.0]K
RXS2 32 0 [10000.0,0.0]K
RXS3 33 0 [10000.0,0.0]K
RXS4 34 0 [10000.0,0.0]K
RSUM 6 7 [500.0,5.0]
RF 8 7 [9258.7,92.5]
X741 7 0 8 10 11 UA741
.op
.end
```

*Figure 2-8C.* SFA input file for D/A converter. Component variation specified as 1 $\sigma$ = 5%.

The output of the SFA routines were analyzed using the SAS DISCRIM function. Figure 2-8D shows typical output from the program for the bit pattern 0111. The fault classes are identified by the code name INX0111, where N refers to which of the five current sources (numbered 0 to 4) is faulted, and X replaces the letter P for "open" or S for "short." Nominal cases are identified by the fault class label 0N0111.

At first glance, it appears as though very poor classification has taken place, and that many of the fault classes become confused among each other and the nominal case. However, well-defined patterns do emerge from the data. In the case of 0111 @ 5% sigma, 94% of the faults where current source 2 was opened were properly classified. This current source is also the most significant "on" bit. Similar highly successful classification rates were obtained for the other bit patterns when the most significant "on" bit's current source was opened. The success of the classification dropped as the current sources corresponding to consecutively lower significant bits were opened. This is because the contribution of the lower significant bits became masked by the inherent variability of the circuit performance. Table 2-8A summarizes the discrimination results.

Another pattern that emerges from the data is the creation of two distinct fault groups. The first group forms an ambiguity set among the nominals and lower significant bit open current sources. The second group forms an ambiguity set containing all the short-circuited cases. The cause of the short circuit grouping is obvious when viewing Figure 2-8A. A short circuit anywhere along the current summation bus (at $V_6$) produces essentially the same effect. The ambiguity group that confuses several open cases and the nominals arises for two reasons. The first reason is bits that are already turned off will not be affected by the presence of an open circuit at the corresponding current source. The second reason was discussed above, where the inherent variability of the circuit masks contributions from the lower significant bits.

Note most of the circuit variability arises from the op-amp portion of the circuit. However, the inherent variation of the current sources themselves can result in bit error. This can be seen by reviewing the nominal currents and their standard deviation over 50 Monte Carlo runs (at parameter variations of 1 sigma = 5%):

| current source | mean (μA) | Std. Dev. (uA) |
|---|---|---|
| I0 | 42.06 | 6.21 |
| I1 | 42.06 | 6.21 |
| I2 | 42.06 | 6.21 |
| I3 | 21.25 | 3.53 |
| I4 | 10.46 | 1.68 |

Since the current sources I0 and I1 are doubled to create the most significant bit, sufficient variation in I0 and I1 can mask I4, resulting in bit error and fault detection ambiguity.

Classification of DA0111 data

Discriminant Analysis  Classification Summary for Calibration Data  WORK.RAWDAT

Resubstitution Summary using Quadratic Discriminant Function

Generalized Squared Distance Function:  Posterior Probability of Membership in each FAULT1

$$D_j(X) = (X-\bar{X}_j)' \, COV_j^{-1} \, (X-\bar{X}_j) + \ln |COV_j| \qquad Pr(j|X) = \exp(-.5\, D_j(X)) \,/\, \sum_k \exp(-.5\, D_k(X))$$

Number of Observations and Percent Classified into FAULT1.

| From FAULT1 | I0P0111 | I1P0111 | I1S0111 | I2P0111 | I2S0111 | I3P0111 | I3S0111 | I4P0111 | I4S0111 | IOS0111 | ON0111 | Total |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| I0P0111 | 8 | 13 | 0 | 0 | 0 | 10 | 0 | 10 | 0 | 0 | 9 | 50 |
| | 16.00 | 26.00 | 0.00 | 0.00 | 0.00 | 20.00 | 0.00 | 20.00 | 0.00 | 0.00 | 18.00 | 100.00 |
| I1P0111 | 5 | 19 | 0 | 0 | 0 | 4 | 0 | 15 | 0 | 0 | 7 | 50 |
| | 10.00 | 38.00 | 0.00 | 0.00 | 0.00 | 8.00 | 0.00 | 30.00 | 0.00 | 0.00 | 14.00 | 100.00 |
| I1S0111 | 0 | 0 | 8 | 0 | 14 | 0 | 7 | 0 | 5 | 16 | 0 | 50 |
| | 0.00 | 0.00 | 16.00 | 0.00 | 28.00 | 0.00 | 14.00 | 0.00 | 10.00 | 32.00 | 0.00 | 100.00 |
| I2P0111 | 0 | 0 | 0 | 47 | 0 | 3 | 0 | 0 | 0 | 0 | 0 | 50 |
| | 0.00 | 0.00 | 0.00 | 94.00 | 0.00 | 6.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 100.00 |
| I2S0111 | 0 | 0 | 1 | 0 | 35 | 0 | 3 | 0 | 0 | 11 | 0 | 50 |
| | 0.00 | 0.00 | 2.00 | 0.00 | 70.00 | 0.00 | 6.00 | 0.00 | 0.00 | 22.00 | 0.00 | 100.00 |
| I3P0111 | 1 | 1 | 0 | 4 | 0 | 34 | 0 | 10 | 0 | 0 | 0 | 50 |
| | 2.00 | 2.00 | 0.00 | 8.00 | 0.00 | 68.00 | 0.00 | 20.00 | 0.00 | 0.00 | 0.00 | 100.00 |
| I3S0111 | 0 | 0 | 4 | 0 | 17 | 0 | 5 | 0 | 2 | 22 | 0 | 50 |
| | 0.00 | 0.00 | 8.00 | 0.00 | 34.00 | 0.00 | 10.00 | 0.00 | 4.00 | 44.00 | 0.00 | 100.00 |
| I4P0111 | 3 | 6 | 0 | 0 | 0 | 16 | 0 | 21 | 0 | 0 | 4 | 50 |
| | 6.00 | 12.00 | 0.00 | 0.00 | 0.00 | 32.00 | 0.00 | 42.00 | 0.00 | 0.00 | 8.00 | 100.00 |
| I4S0111 | 0 | 0 | 4 | 0 | 14 | 0 | 8 | 0 | 4 | 20 | 0 | 50 |
| | 0.00 | 0.00 | 8.00 | 0.00 | 28.00 | 0.00 | 16.00 | 0.00 | 8.00 | 40.00 | 0.00 | 100.00 |
| IOS0111 | 0 | 0 | 3 | 0 | 8 | 0 | 5 | 0 | 2 | 32 | 0 | 50 |
| | 0.00 | 0.00 | 6.00 | 0.00 | 16.00 | 0.00 | 10.00 | 0.00 | 4.00 | 64.00 | 0.00 | 100.00 |
| ON0111 | 7 | 16 | 0 | 0 | 0 | 7 | 0 | 8 | 0 | 0 | 12 | 50 |
| | 14.00 | 32.00 | 0.00 | 0.00 | 0.00 | 14.00 | 0.00 | 16.00 | 0.00 | 0.00 | 24.00 | 100.00 |
| Total | 24 | 55 | 20 | 51 | 88 | 74 | 28 | 64 | 13 | 101 | 32 | 550 |
| Percent | 4.36 | 10.00 | 3.64 | 9.27 | 16.00 | 13.45 | 5.09 | 11.64 | 2.36 | 18.36 | 5.82 | 100.00 |
| Priors | 0.0909 | 0.0909 | 0.0909 | 0.0909 | 0.0909 | 0.0909 | 0.0909 | 0.0909 | 0.0909 | 0.0909 | 0.0909 | |

DA -- Digital Analog Converter

*Figure 2-8D.* Typical SAS output for analysis of D/A converter. Print-out for the bit pattern 0111 is shown.

# Table 2-8A. Summary of D/A Converter Fault Classification

| *pattern* | *sigma* | *distinguished groups* | *ambiguity sets* | |
|---|---|---|---|---|
| | | | *set 1* | *set 2* |
| 0000 | 5% | none | all shorts | all opens and nominals |
| 0000 | 2.5% | none | " | " |
| 0001 | 5% | I4 open (100%)[1] | all shorts | all nominals and remaining opens |
| 0001 | 2.5% | I4 open (100%) | " | " |
| 0011 | 5% | I3 open (96%)<br>I4 open (86%) | " | " |
| 0011 | 2.5% | I3 open (100%)<br>I4 open (100%) | " | " |
| 0111 | 5% | I2 open (94%)<br>I3 open (68%)<br>I4 open (42%) | " | " |
| 0111 | 2.5% | I2 open (100%)<br>I3 open (92%)<br>I4 open (76%) | " | " |
| 1111 | 5% | I0 open (36%)<br>I1 open (32%) | " | " |
| 1111 | 2.5% | I0 open (34%)<br>I1 open (34%)<br>I2 open (34%)<br>I3 open (56%)<br>I4 open (48%) | " | " |

[1]Percentage of simulated faults properly classified over 50 Monte Carlo runs. Training set and test data were equivalent.

## 2.9. Distributed Amplifier

### 2.9A. Circuit Description

The distributed amplifier used in this study is based on a design constructed at the David Sarnoff Research Center. The design calls for the use of monolithic microwave integrated circuit technology and contains four FETs connected in a travelling wave configuration (see Figures2-9A and 2-9B). SPICE simulations made use of the Statz-Raytheon MESFET model, as implemented in UC Berkeley's SPICE 3C1. The frequency range of the simulations spanned 1 to 20 GHz. A representative SFA input file gives further details of the circuit (Figure 2-9C).

### 2.9B. Fault Analysis

The fault analysis was confined to each of the possible opens or shorts at each of the four FETs. However, no more than one short or open could simultaneously exist. Data used for the analysis were the DC gate and drain voltages of the devices, as well as the overall AC output voltage from the circuit. These voltages were selected because they are easily accessible in real measurements. Note that because the circuit operates at microwave frequencies, AC measurements of the internal circuit nodes are generally difficult and impractical to obtain because of probe loading effects (unless newer methods, such as optical probing, are used). Device and circuit resistances and capacitances were given standard deviations $\sigma$ of 1%, 5%, and 10% for nominal variations. A standard deviation of 1% variation was assigned to the transmission line parameters throughout the analysis.

As in past circuits explored in our work, we applied both the discrimination analysis and hypothesis testing methodologies. 100 Monte Carlo runs for each fault type trained the discriminator, and 100 Monte Carlo runs were used to test the discriminator and hypothesis testing. Training and testing made use of the same data.

The results were very encouraging. For the $\sigma$ = 10% case, Table 2-9A shows the results of the fault classification and hypothesis tests. Note, that the hypothesis test, i.e., the test of whether or not a circuit falls into the nominal category or alternatively does not, is not nearly as effective as

the classification process. This is because during the construction of the classification rules the distributions of the fault classes are used to partition the parameter space according to the possible faults that can occur. However, during the hypothesis test only the distribution of the nominal circuits is considered. It happens that the 95% contour of the nominal circuit distribution overlaps some of the faulted circuit distributions. Consequently there are several faults which are misclassified as nominal (see Figure 2-9D). However, because the discrimination takes into account each of the fault distributions, fault misclassification rarely occurs using the classification rules.

For the data from $\sigma = 5\%$ and 1% cases, all faulted and nominal circuits were correctly classified and all faults were rejected using the hypothesis test. This is quite an encouraging result.

## Table 2-9A

**Summary of Discrimination and Hypothesis Test Results**
for $\sigma = 10\%$ component variation

| | Fault Class | % Correctly Classified *(discriminant analysis)* | % Correctly Detected *(hypothesis test)* |
|---|---|---|---|
| 1 | Nominal | 90% | 93% |
| 2 | RD1_Open | 100 | 70 |
| 3 | RS1_Open | 100 | 100 |
| 4 | RG1_Open | 100 | 100 |
| 5 | GS1_Short | 100 | 100 |
| 6 | DS1_Short | 100 | 100 |
| 7 | GD1_Short | 100 | 100 |
| 8 | RD2_Open | 99 | 100 |
| 9 | RS2_Open | 89 | 70 |
| 10 | RG2_Open | 100 | 100 |
| 11 | GS2_Short | 100 | 100 |
| 12 | DS2_Short | 100 | 100 |
| 13 | GD2_Short | 100 | 100 |
| 14 | RD3_Open | 99 | 100 |
| 15 | RS3_Open | 94 | 80 |
| 16 | RG3_Open | 97 | 100 |
| 17 | GS3_Short | 100 | 100 |
| 18 | DS3_Short | 100 | 100 |
| 19 | GD3_Short | 99 | 100 |
| 20 | RD4_Open | 98 | 100 |
| 21 | RS4_Open | 88 | 70 |
| 22 | RG4_Open | 99 | 100 |
| 23 | GS4_Short | 100 | 100 |
| 24 | DS4_Short | 100 | 100 |
| 25 | GD4_Short | 100 | 100 |
| | | $\sigma = 5\%$ component variation | |
| 1 | Nominal | 100% | 94% |
| 2 | All faults | 100 | 100 |
| | | $\sigma = 1\%$ component variation | |
| 1 | Nominal | 100% | 98% |
| 2 | All faults | 100 | 100 |

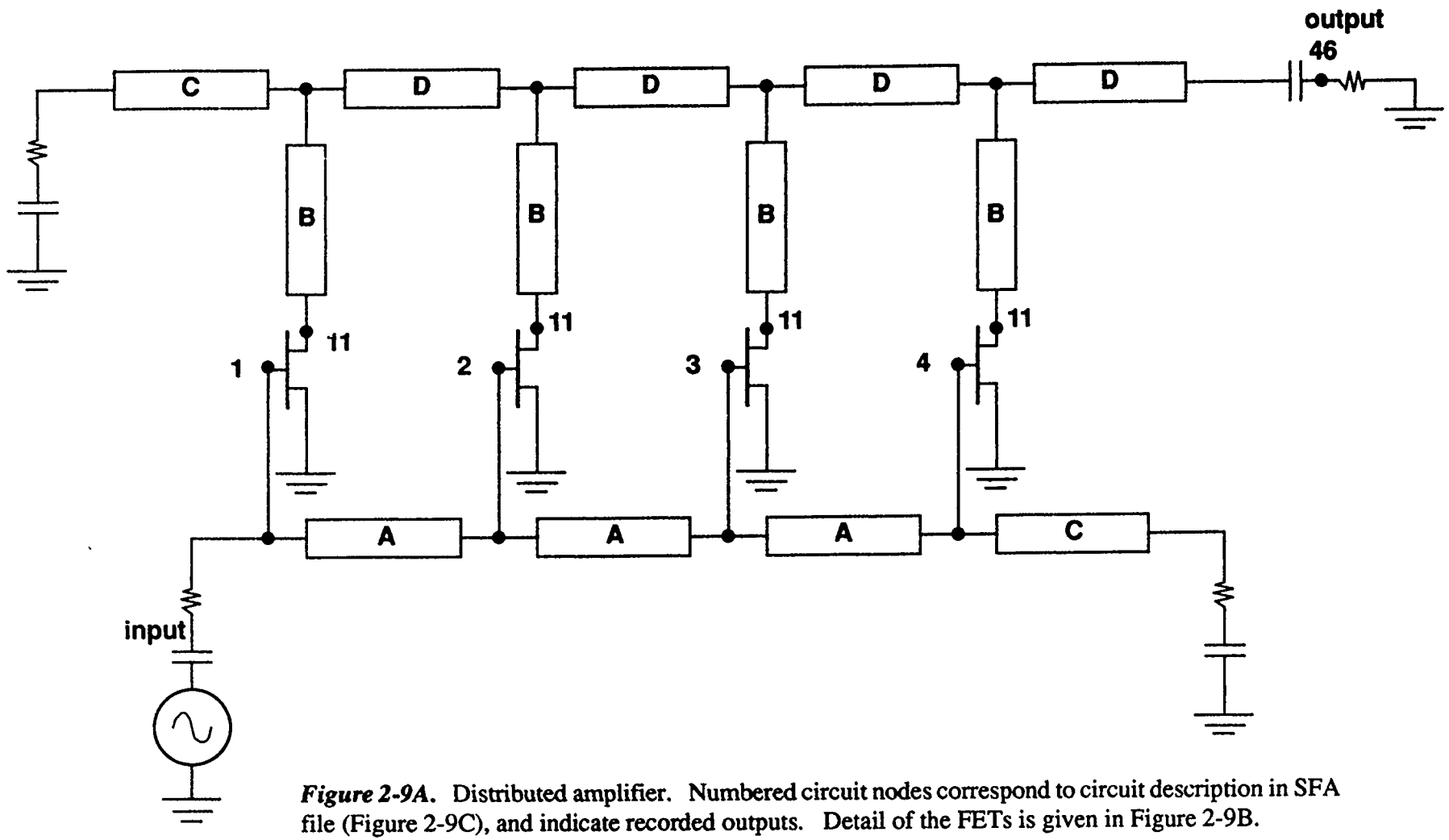



*Figure 2-9A.* Distributed amplifier. Numbered circuit nodes correspond to circuit description in SFA file (Figure 2-9C), and indicate recorded outputs. Detail of the FETs is given in Figure 2-9B.

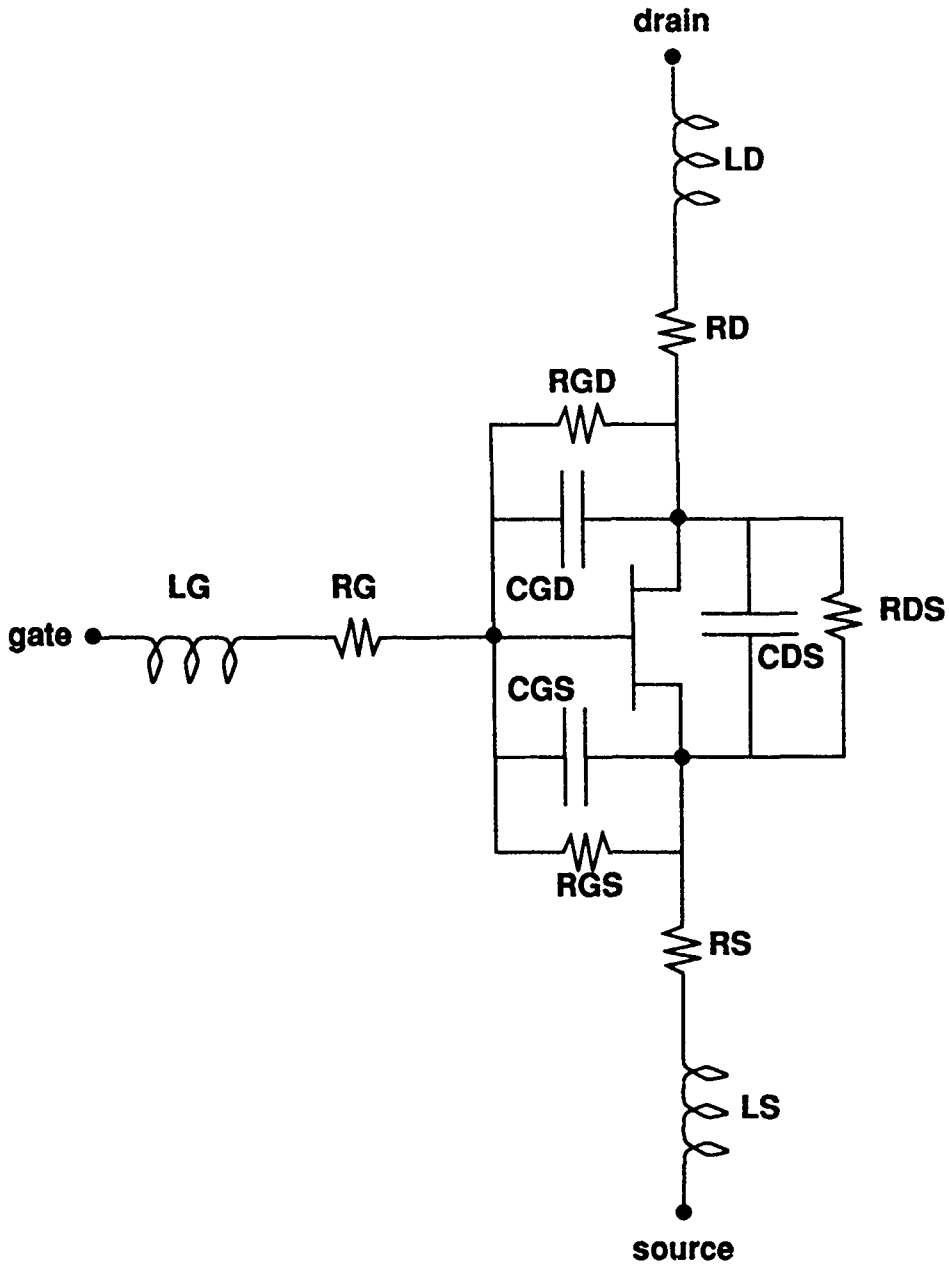



***Figure 2-9B.*** Detail of the FETs in distributed amplifier of Figure 2-9A. All resistances are varied during the analysis to invoke catastrophic faults.

```
Distributed Amplifier
* B. Epstein  8/90
* sigma = 5% for all component values
* sigma = 1% for all dimensions
* Nominal Circuit:
#GOOD 100
*
* FET1 Faults
#RD1 1.0e6 100 RD1_OPEN
#RS1 1.0e6 100 RS1_OPEN
#RG1 1.0e6 100 RG1_OPEN
#RGS1 0.001 100 GS1_SHORT
#RDS1 0.001 100 DS1_SHORT
#RGD1 0.001 100 GD1_SHORT
*
* FET2 Faults
#RD2 1.0e6 100 RD2_OPEN
#RS2 1.0e6 100 RS2_OPEN
#RG2 1.0e6 100 RG2_OPEN
#RGS2 0.001 100 GS2_SHORT
#RDS2 0.001 100 DS2_SHORT
#RGD2 0.001 100 GD2_SHORT
*
* FET3 Faults
#RD3 1.0e6 100 RD3_OPEN
#RS3 1.0e6 100 RS3_OPEN
#RG3 1.0e6 100 RG3_OPEN
#RGS3 0.001 100 GS3_SHORT
#RDS3 0.001 100 DS3_SHORT
#RGD3 0.001 100 GD3_SHORT
*
* FET4 Faults
#RD4 1.0e6 100 RD4_OPEN
#RS4 1.0e6 100 RS4_OPEN
#RG4 1.0e6 100 RG4_OPEN
#RGS4 0.001 100 GS4_SHORT
#RDS4 0.001 100 DS4_SHORT
#RGD4 0.001 100 GD4_SHORT
*
%POLAR
%dc 1 2 3 4 11 12 13 14
%ac 46
.SUBCKT FET1 1 2 3
LD1 1 11 0.04nH
RD1 11 21 [2.0,0.2]
LG1 2 12 0.065nH
RG1 12 22 [1.0,0.1]
LS1 3 13 0.025nH
RS1 13 23 [1.0,0.1]
RGS1 22 23 [10000.0,0.0]K
RDS1 21 23 [10000.0,0.0]K
RGD1 21 22 [10000.0,0.0]K
CGS1 22 23 [0.55,0.055]pF
CDS1 21 23 [0.12,0.012]pF
CGD1 21 22 [0.025,0.0025]pF
Z1 21 22 23 MES1
.MODEL    MES1    NMF    (VTO=[-2.66,0.027]
BETA=[0.3239,0.032] B=[3.527,0.35]
+  ALPHA=[3.0,0.3]  LAMBDA=[0.0475,0.0048]
PB=0.6)
.ENDS XZ1
*
.SUBCKT FET2 1 2 3
LD2 1 11 0.04nH
RD2 11 21 [2.0,0.2]
LG2 2 12 0.065nH
RG2 12 22 [1.0,0.1]
LS2 3 13 0.025nH
RS2 13 23 [1.0,0.1]
RGS2 22 23 [10000.0,0.0]K
RDS2 21 23 [10000.0,0.0]K
RGD2 21 22 [10000.0,0.0]K
CGS2 22 23 [0.55,0.055]pF
CDS2 21 23 [0.12,0.012]pF
CGD2 21 22 [0.025,0.0025]pF
Z2 21 22 23 MES2
.MODEL    MES2    NMF    (VTO=[-2.66,.027]
BETA=[0.3239,0.032] B=[3.527,.35]
+ ALPHA=[3.0,0.3]  LAMBDA=[0.0475,0.0047]
PB=0.6)
.ENDS FET2
*
.SUBCKT FET3 1 2 3
LD3 1 11 0.04nH
RD3 11 21 [2.0,0.2]
LG3 2 12 0.065nH
RG3 12 22 [1.0,0.1]
LS3 3 13 0.025nH
RS3 13 23 [1.0,0.1]
RGS3 22 23 [10000.0,0.0]K
RDS3 21 23 [10000.0,0.0]K
RGD3 21 22 [10000.0,0.0]K
CGS3 22 23 [0.55,0.055]pF
CDS3 21 23 [0.12,0.012]pF
CGD3 21 22 [0.025,0.0025]pF
Z3 21 22 23 MES3
.MODEL    MES3    NMF    (VTO=[-2.66,.27]
BETA=[0.3239,0.032] B=[3.527,.35]
+ ALPHA=[3.0,0.3]  LAMBDA=[0.0475,0.0047]
PB=0.6)
.ENDS FET3
*
.SUBCKT FET4 1 2 3
LD4 1 11 0.04nH
RD4 11 21 [2.0,0.2]
LG4 2 12 0.065nH
RG4 12 22 [1.0,0.1]
LS4 3 13 0.025nH
RS4 13 23 [1.0,0.1]
RGS4 22 23 [10000.0,0.0]K
RDS4 21 23 [10000.0,0.0]K
RGD4 21 22 [10000.0,0.0]K
CGS4 22 23 [0.55,0.055]pF
CDS4 21 23 [0.12,0.012]pF
CGD4 21 22 [0.025,0.0025]pF
Z4 21 22 23 MES4
.MODEL    MES4    NMF    (VTO=[-2.66,.27]
BETA=[0.3239,0.032] B=[3.527,.35]
+ ALPHA=[3.0,0.3]  LAMBDA=[0.0475,0.0047]
PB=0.6)
.ENDS FET4
*
```

**Figure 2-9C.** SFA input file for 4 transistor distributed amplifier

```
* Devices:
XZ1 11 1 0 FET1
XZ2 12 2 0 FET2
XZ3 13 3 0 FET3
XZ4 14 4 0 FET4
*
* Transmission Lines:
TA1 1 0 2 0 Z0=[62.3,.6] TD=[0.689,.007]ps
TA2 2 0 3 0 Z0=[62.3,.6] TD=[0.689,.007]ps
TA3 3 0 4 0 Z0=[62.3,.6] TD=[0.689,.007]ps
TA4 4 0 5 0 Z0=[62.3,.6] TD=[0.689,.007]ps
TB1 7 0 11 0 Z0=[50.0,.5] TD=[0.704,.007]ps
TB2 8 0 12 0 Z0=[50.0,.5] TD=[0.704,.007]ps
TB3 9 0 13 0 Z0=[50.0,.5] TD=[0.704,.007]ps
TB4 10 0 14 0 Z0=[50.0,.5] TD=[0.704,.007]ps
TC1 4 0 5 0 Z0=[81.6,.8] TD=[1.787,.017]ps
TC2 7 0 6 0 Z0=[81.6,.8] TD=[1.787,.017]ps
TD1 7 0 8 0 Z0=[80.0.8] TD=[1.61,.016]ps
TD2 8 0 9 0 Z0=[80.0,.8] TD=[1.61,.016]ps
TD3 9 0 10 0 Z0=[80.0,.8] TD=[1.61,.016]ps
TD4 10 0 11 0 Z0=[80.0,.8] TD=[1.61,.016]ps
TIN 25 0 1 0 Z0=[50.0,.5] TD=[0.5,.005]ps
*
* Terminations:
RG 5 35 [10.0,1.0]
CG 35 0 [10.0,1.0]pf
RD 6 36 [30.0,3.0]
CD 36 0 [10.0,1.0]pf
COUT 11 46 [5.0,.5]pF
ROUT 46 0 [50.0,5.0]
*
* DC Bias:
LBI1 40 25 10.0mH
LBI2 41 11 10.0mH
VG 40 0 -1.0
VD 41 0 6.0
*
* Signal:
VGEN 44 0 AC 2.0
RIN 44 45 [25.0,2.5]
CIN 45 25 [1.0,0.1]pF
*
* Control:
.op
.ac lin 5 1.0G 20.0G
.end
```

**Figure 2-9C (cont).** SFA input file for 4 transistor distributed amplifier

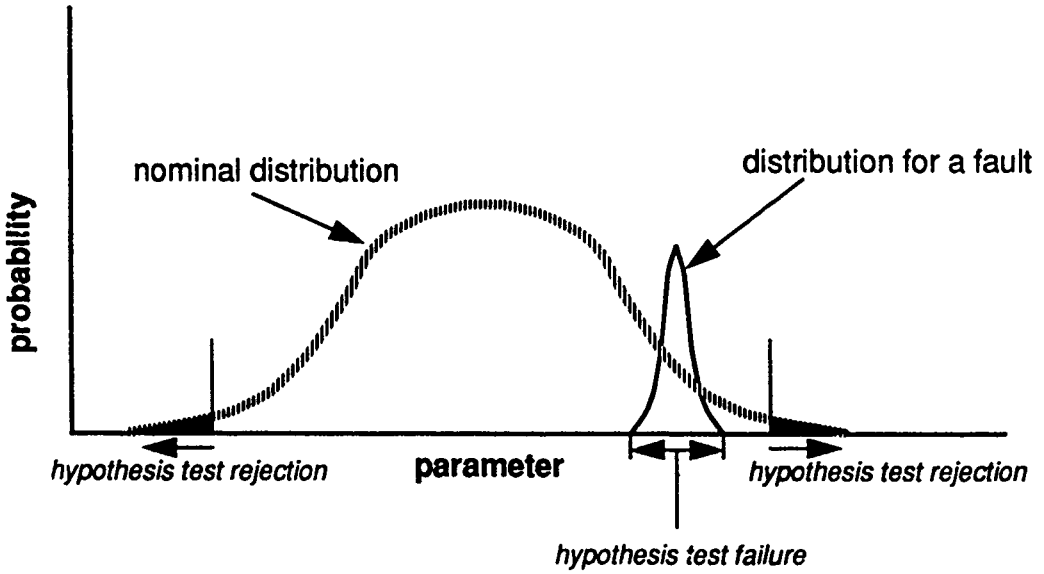



***Figure 2-9D.*** Hypothesis test failure. The curves are the probability distributions of a given measurement parameter (e.g., a node voltage) for the nominal circuit and a faulted case. The hypothesis test fails when a faulty circuit's distribution lies within the 95% region of the nominal circuit distribution, as shown. The hypothesis test only rejects circuits outside of the 95%; i.e., circuits whose response lies within the shaded region of the nominal curve (and beyond), which corresponds in this example to a level of significance of 5%.

## 2.10. Summary of Examples

For the most part, the example circuits investigated during this study yielded results that encourage further consideration of the use of classification techniques for identifying faults in analog microcircuits. Table 2-10A provides a summary of the results. Five of the nine circuits yielded APERs of less than 1%. Those circuits with higher APERs deserve further analysis (i.e., different measurement nodes, operating conditions, etc.) to reduce their error rates. Likewise, other classification schemes should be investigated to improve the classification and fault detection power of the tests. One promising means of classification employes neural networks [20].

Although the examples make fault detection and classification promising, one major caveat persists -- the *fault detection and classification methods must be performed using real IC data!* Real data will reflect the true variability of circuit performance and component behavior, and violate many of the implicit assumptions of the analyses performed in this study (e.g., the assumption of normality, accuracy of the component models and simulations). The work described in this report sets the stage for such a study.

***Explanation of Table 2-10A (next page):***

1. "Measurement" refers to simulated voltage data
2. APER is the apparent rate of error (see Section 1.4).
3. 2-class discrimination analysis and hypothesis tests performed Go/No-Go testing.
4. Refer to Section 2.8 for explanation of high APER.

**Table 2-10A. Summary of Results**

| Circuit Type | Measurement Nodes[1] | # Fault Classes | # Points | APER[2] | Comment |
|---|---|---|---|---|---|
| One-stage amplifier | 3 nodes, DC only | 7 | 7000 | < 0.2% | internal device nodes included |
| | 4 nodes, DC only | 7 | 1400 | 22% | external device nodes only |
| | 4 nodes, AC and DC | 7 | 1400 | 0.3% | external device nodes only |
| Differential pair | 6 nodes, DC only | 33 | 6600 | 23 to 24% | nearest k neighbor |
| | | | | 18 to 24% | discrimination analysis |
| Macro op-amp | 2 nodes, AC and DC | 2 | 200 | 1.4 to 3.3% | 2-class discrimination analysis[3] |
| | | | | 6.8% | hypothesis test[3] |
| Elliptical filter | 3 nodes, AC freq. sweep & DC | 23 | 640 | 0.62% | discrimination analysis |
| | | 23 | 222 | 1.8% | 2-class discrimination analysis |
| | | 23 | 62 | 0% | hypothesis test |
| Comparator | 1 node, DC ramped | 7 | 700 | 2.4% | discrimination analysis |
| | | 7 | 700 | 0.86% | hypothesis test |
| Logarithmic amp | 3 nodes, all DC ramped | 25 | 440 | 9.1% | discrimination analysis |
| | | 25 | 440 | 14.5% | hypothesis test |
| Analog multiplexer | 10 nodes, DC only | 15 | 240 | 7.9% | discrimination analysis |
| | | 15 | 240 | 35% | nearest k neighbor |
| | | 15 | 240 | 18.3% | hypothesis test |
| D/A converter | 6 nodes, DC only | 11 | 550 | 0 to 68%[4] | discrimination analysis |
| Distributed amplifier | 8 DC nodes, 1 AC node | 25 | 2500 | 1.9% | discrimination analysis, $\sigma$=10% |
| | | 25 | 2500 | 0 | discrimination analysis, $\sigma$=5% |
| | | 25 | 2500 | 0 | discrimination analysis, $\sigma$=1% |
| | | 25 | 2500 | 4.7% | hypothesis test, $\sigma$=10% |
| | | 25 | 2500 | 0.24% | hypothesis test, $\sigma$=5% |
| | | 25 | 2500 | <0.1% | hypothesis test, $\sigma$=1% |

*see previous page for explanation of table*

# References


1. G.R. Boyle, B.M. Cohn, D.O. Pederson, J.E. Solomon, "Macromodeling of Integrated Circuit Operational Amplifiers," *IEEE J. Solid State Circuits*, vol. SC-9, no. 6, Dec. 1974, pp. 353-364.

2. W.G. Jung, *IC Op-Amp Cookbook*, Howard W. Sams & Co., Indianapolis, 1974.

3. J.V. Wait, L.P. Huelsman, and G.A. Korn, *Introduction to Operational Amplifier Theory and Applications*, McGraw-Hill, New York, 1975.

4. L.W. Nagel, "SPICE2: A computer program to simulate semiconductor circuits," Ph.D. Dissertation, Electronics Research Laboratory, Univ. of CA, Berkeley, 1975. See also T.L. Quarles, *SPICE3 Version 3C1 User's Guide* (memorandum No. UCB/ERL M89/46; April 1989) and associated documentation from Univ. of CA - Berkeley's Electronics Research Laboratory,

5. *IEEE Trans. on Circuits and Systems*, (Special issue on automatic analog fault diagnosis), vol. CAS-26, no. 7, July 1979.

6. K.C. Varghese, J.H. Williams, and D.R. Towill, "Simplified ATPG and analog fault location via a clustering and separability technique," *IEEE Trans. on Circuits and Systems*, CAS-26, no. 7, July 1979, pp. 496-505.

7. W. Hochwald and J.D. Bastian, "A DC approach for analog fault dictionary determination," *IEEE Trans. on Circuits and Systems*, CAS-26, no. 7, July 1979, pp. 523-529.

8. J.J. Moré, B.S. Garbow, and K.E. Hillstrom, *User Guide to MINPACK-1*, Argonne National Laboratory, August, 1980.

9. D.E. Knuth, *Seminumerical Algorithms* (Vol. 2 of *The Art of Computer Programming*), Addison-Wesley, Reading Mass, 1981.

10. M. Kendall, A. Stuart, and J.K. Ord, *The Advanced Theory of Statistics*, vol. 3, Charles Griffin and Co., London, 1983, pp. 267-317.

11. J.W. Bandler and A.E. Salama, "Fault diagnosis of analog circuits," *Proceedings IEEE*, vol. 73, no. 8, pp. 1270-1325, August 1985.

12. N.R. Draper, "Plackett and Burman Designs," in *Encyclopedia of Statistical Sciences, Vol. 6*, S.Kotz and N.L. Johnson, eds., John Wiley and Sons, New York, 1985, pp. 754-758

13. *SAS User's Guide: Statistics*, SAS Institute Inc., Box 8000, Cary NC 27511-8000, 1985.

14. *RLA Series Linear Array Design Manual*, Raytheon Company Semiconductor Division, 2nd Edition, 1987.

15. *Selected Papers on Analog Fault Diagnosis*, R-W Liu, ed., IEEE Press, New York, 1987.

16. R.A. Johnson and D.W. Wichern, *Applied Multivariate Statistical Analysis*, Prentice Hall, Englewood Cliffs, N.J., 1988, pp. 120-209.

17. L.G. Meares and C.E. Hymowitz, *Simulating with SPICE*, Intusoft, San Pedro, CA, 1988.

18. L. Milor and V. Visvanathan, "Detection of catastrophic faults in analog integrated circuits," *IEEE Trans. Computer-Aided Design*, vol. 8, no. 2, Feb. 1989, pp. 114-130.

* 19. B.R. Epstein, *Linear Microcircuit Fault Modeling and Simulation*, Interim Report, Rome Air Development Center, contract F30602-89-0076, April 1990.

20. B.R. Epstein, M.H. Czigler, and S.R. Miller, "Linear Microcircuit Fault Modeling and Detection," *Digest of the 1991 IEEE VLSI Test Symposium*, Atlantic City, April 1991.

21. Math Soft, Inc. 201 Broadway, Cambridge, MA, 02139.



*Although this report references the * limited document listed above, no limited information has been extracted, DOD and DOD contractors only; premature dissemination; Apr 90.

# Appendix: Using the Statistical Fault Analyzer (SFA)

The SFA has two modes of operation. The first mode generates a training set file, where the file is later used during an analysis by SAS (or other statistical analysis package). The second mode performs classification and hypothesis testing of simulated test data. Both modes use a common input file based on SPICE syntax. Refer to Section 1.4 for further information on the overall program flow.

## A.1. Input File

The construction of the an input file is best understood by example. Figure A-1 shows a file for the one-stage amplifier, which was analyzed extensively in Section 2.1. (*NOTE:* the line numbers at the beginning of each line in the Figure A-1 are not to be inserted in the file; they only have been added for clarity.) SPICE users will immediately notice that much of the file follows standard SPICE syntax, with the exception of a few added lines and symbols. *All users should become familiar with SPICE before using the SFA!*

### 1) Comment Lines

The first line of the file can be any text string. Use this line to identify the purpose of the file. All subsequent "comment lines" in the file must begin with the *symbol (e.g., lines 4, 15, 20, 21).

### 2) Fault Specification Lines

Fault specifications begin with the # symbol (lines 5 to 14). Going across a line beginning with #, the following information is supplied:

*For SPICE passive components:*

Component name (immediately following the # sign)
Component value when component is faulted
The number of Monte Carlo simulations to be performed while the component is faulted
A user-specified one-word text label identifying the fault.

*For SPICE active device models:*

The above four parameters plus the name of the SPICE model parameter to be faulted.

```
1    Single-stage amplifier
2    * Nominal Operation:
3    #GOOD 200
4    * Faults:
5    #RC 1.0e6 200 RC_OPEN
6    #RE 1.0e6 200 RE_OPEN
7    #RB 1.0e6 200 RB_OPEN
8    #RB 1.0 200 RB_Short
9    #Rbc 1.0 200 RBC_Short
10   #Rbe 1.0 200 RBE_Short
11   #Rce 1.0 200 RCE_Short
12   #R3 0.001 200 R1_Short
13   #QNL 1.0 200 Q_low BF
14   #QNL 200.0 200 Q_hi BF
15   * Statistical Analysis Nodes:
16   %dc 1 2 3 7
17   %ac 2 3 7
18   %TRACE
19   %POLAR
20   *
21   * Circuit:
22   VCC 8 0 5.0
23   RSDC 8 2 1.0
24   VIN 10 0 AC 1
25   CIN 10 1 100.0UF
26   R3 2 3 [1.2,.06]K
27   RC 3 6 [1.0,0.0]
28   RB 1 5 [100.0,5.0]
29   R2 2 1 [61.0,3.1]K
30   R1 1 0 [17.6,0.9]K
31   RE 4 7 [1.0,0.0]
32   R4 7 0 [300.0,15.0]
33   Rbc 5 6 [1.0e8,0.0]
34   Rbe 5 4 [1.0e8,0.0]
35   Rce 6 4 [1.0e8,0.0]
36   Q1 6 5 4 QNL
37   .op
38   .AC LIN 1 100 100hz
39   .model QNL NPN(BF=[80,12] CCS=2PF TF=0.3NS TR=6NS CJE=3PF
40   + CJC=2PF VA=[50,7.5])
41   .END
```

***Figure A-1.*** Example SFA input file for one-stage amplifier.

Use the component name as it appears in the SPICE circuit description. For example, line 7 states that resistor **RB** is to be faulted. Resistor **RB** is defined on line 28. On line 7, resistor **RB** acquires the value of 1 megΩ when it is faulted. The value specified replaces the square brackets, and the quantities between them, in the SPICE specification (line 28 in this case). Any units specified outside of the brackets on the SPICE line are preserved, even during the fault simulation. Therefore, a faulted value of 0.001 for **R3** (line 12) takes on the value of 0.001K, since in line 26 the unit of K (for kilo-ohms) is present.

The number of Monte Carlo iterations may be any positive integer starting at 1. In this example, all faults have 200 iterations; however, different faults can have different iteration counts. The text label is used by SAS (and other programs) to specify the class name for each fault. This label is also used to identify and display fault types when SFA operates in the test mode.

Note active device models include the name of the SPICE parameter to be modified when the component is faulted. In this example the active device, **QNL**, is used only once. However, when there are multiple instances of the active component, its modified value will be used wherever that component is instanced.

Users can take full advantage of SPICE3C1's submodeling capability. Components within submodels are faulted in the manner described above (for example, see Figure 2-9C).

The line **#GOOD 200** (line 3) states that 200 simulations of nominal circuit operation are to be performed. No component value or identification label is specified in this line.

During SFA operation, each fault is activated one at a time for the specified number of Monte Carlo simulations. When the specified number of simulations at a given fault is completed, the next fault in the list is executed.

### 3) Nominal and Standard Deviation Specification

These parameters are specified within the squared brackets [ ] for all nondeterministic components in the SPICE file. The first value is the mean component value, and the second parameter is the 1 sigma standard deviation (e.g., see lines 26 to 35). The unit specification outside the square bracket

is preserved. When in operation, the SFA effectively replaces the bracketed quantities with values that vary randomly from Monte Carlo iteration to iteration.

All components that are to be faulted *must also have a square bracket specification*, even if such a component is not to vary randomly. In the latter case, use 0.0 as the standard deviation (e.g., lines 33-35).

### 4) Analysis Outputs

DC nodal voltages to be sent to the training set file are specified by the **%dc** line. In the example, line 16 states that DC nodal voltages at nodes 1, 2, 3, and 7 are to be stored. AC phasor voltages are stored in the same manner; however, the real and imaginary part for each node in the **%ac** line are now stored. If the user desires that the phasor voltages be stored by magnitude (in dB) and angle (in degrees), use the key word **%POLAR** (line 19). If **%POLAR** is not specified, the AC phasors are stored as real and imaginary parts. Multiple **%dc** and **%ac** lines may be used when all node numbers cannot fit on one line.

### 5) Tracing

As explained in Section 1.4, the SFA generates a standard SPICE input file to be read by SPICE during each Monte Carlo iteration. The SFA replaces all values between the square brackets with faulted and random component values when constructing the SPICE input files. The user may observe these files on the screen, just prior to each SPICE simulation, by including the key word **%TRACE** (line 18). When **%TRACE** is not specified, a series of dots will trail across the screen while SFA operates. Each dot represents one Monte Carlo iteration; hence, one SPICE execution.

### 6) File Name

Name all SFA input files using the extension **.cat**. For example, the circuit of Figure A-1 has the name **samp.cat** .

## A.2. Output Files

The SFA program generates two files, both using the name of the input file, but with its **.cat** extension replaced by either **.key** or **.sas** .

### 1) Data File

The data file contains the numerical results of the simulations, which are the DC and AC nodal voltages indicated on the **%dc** and **%ac** lines of the input file, respectively. This file can become quite large when there are many Monte Carlo iterations for each fault, or when outputs are frequency or voltage swept. This file is automatically named by the SFA, and takes on the input file name followed with the extension **.sas** . Each field in the file is the fault label (specified on the **#** lines of the input file), followed by the nodal voltages. All voltages (AC and DC) are in the units of volts, except when the input file contains the **%POLAR** keyword. In this case, the AC voltages are specified in DC and degrees. Figure 1-9 shows a sample **.sas** file.

### 2) Key File

This file tells the user the meaning of each data column in the data output file. The key file name uses the input file name, followed by the **.key** extension. Three columns are contained in this file. The first column is the column number in the data output data file. The second and third columns specify the node and the type of voltage, respectively. For example, the term **operating** refers to DC operating point. **acan real** and **acan imag** refer to real and imaginary parts of the AC analysis voltages, respectively. In general, labeling terms contained in the *rawspice* (c.f. Section 1.4) file are what appear as identifiers.

## A.3 Program Execution

The SFA program, as shipped with this report, has the name ***bsct***. The software is intended to operate on most SUN Microsystems workstations, including the series 3, 4, and SPARCstation computers.

### 1) Training Set Generation

To generate a training set with bsct, simply enter

```
bsct
```

The prompt

```
Enter name of training set file
(or SKIP to skip statistical analysis)
```

will appear. Type in **skip** (upper or lower case), which will tell bsct that testing is not to be performed. The prompt

```
Enter SPICE input file:
```

then appears. Enter the name of the SFA input file you wish to analyze. Do not forget the **.cat** extension. bsct then reads in the file, and reports any errors. With no errors present, the SPICE executions begin for each fault type. A dot will appear on the screen for each Monte Carlo iteration. In the event that SPICE encounters an error (usually due to a convergence problem), a message will appear on the screen and continue with the next Monte Carlo iteration. Erroneous data, in general, will not reach the **.sas** output data file when such a message appears. However, *the user must be aware that even when no error message is reported, SPICE can nevertheless generate strange results!* All users should carefully review their results before proceeding too far with the statistical analyses.

### 2) Testing

bsct performs testing using the procedures described above. However, following the prompt

```
Enter name of training set file
(or SKIP to skip statistical analysis)
```

enter the name of a training set file. Such files are generated following the above procedures. When the prompt

```
Enter SPICE input file:
```

appears, the SFA input file to be entered (with the **.cat** extension) now describes the circuit faults to be tested.

After successfully reading in all files, bsct then computes the covariance matrices and other quantities for each fault type and the pooled data from the faults. SPICE simulations for each fault type

then occur in a manner identical to that when the training set data was generated. While testing, bsct informs the user of the fault type under test, and the success of its classification and hypothesis test. When testing of all fault types is completed, a summary of the results is displayed.

### 3) Operating Options

When operating bsct in the training set or test mode, the random number generator starts of with the default seed value of -1. This can be overridden by use of the command option **s**, followed by a user-specified seed integer. For example, the command

```
bsct s 3
```

causes bsct to use a starting seed value of 3. Use different starting seed values when new output data sets are to represent the same simulations, but with different random values for the components. For example, when the training set and test data sets are to be different, this can be done so by specifying different starting seeds.

The runtime option c, invoked as **bsct c** prints out the covariance matrices and other information as training set data is read into bsct during testing. The runtime option **h**, as in **bsct h**, instructs bsct to skip the classification and perform hypothesis testing only when bsct runs in the test mode. All options can be specified at once; however, make sure that the **s** option is last. For example,

```
bsct cs -4
```

tells bsct to print out all covariance matrices, and define the starting random number seed value as -4.

***MISSION***

***OF***

***ROME LABORATORY***

*Rome Laboratory plans and executes an interdisciplinary program in research, development, test, and technology transition in support of Air Force Command, Control, Communications and Intelligence ($C^3I$) activities for all Air Force platforms. It also executes selected acquisition programs in several areas of expertise. Technical and engineering support within areas of competence is provided to ESD Program Offices (POs) and other ESD elements to perform effective acquisition of $C^3I$ systems. In addition, Rome Laboratory's technology supports other AFSC Product Divisions, the Air Force user community, and other DOD and non-DOD agencies. Rome Laboratory maintains technical competence and research programs in areas including, but not limited to, communications, command and control, battle management, intelligence information processing, computational sciences and software producibility, wide area surveillance/sensors, signal processing, solid state sciences, photonics, electromagnetic technology, superconductivity, and electronic reliability/maintainability and testability.*